



TUGAS AKHIR - KI141502

**RANCANG BANGUN APLIKASI MEDIA INFORMASI
NUTRISI PADA MAKANAN ATAU PRODUK
MAKANAN MENGGUNAKAN TEKNOLOGI
AUGMENTED REALITY BERBASIS ANDROID**

**EKO ADHI WIYONO
NRP 5111100002**

**Dosen Pembimbing I
Wijayanti Nurul Khotimah, S.Kom., M.Sc.**

**Dosen Pembimbing II
Ridho Rahman Hariadi, S.Kom., M.Sc.**

**JURUSAN TEKNIK INFORMATIKA
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2015**



FINAL PROJECT - KI141502

DESIGN AND IMPLEMENTATION OF FOODS AND FOOD PRODUCTS NUTRITION USING AUGMENTED REALITY TECHNOLOGY IN ANDROID

**EKO ADHI WIYONO
NRP 5111100002**

**Advisor I
Wijayanti Nurul Khotimah, S.Kom., M.Sc.**

**Advisor II
Ridho Rahman Hariadi, S.Kom., M.Sc.**

**INFORMATICS DEPARTMENT
Faculty of Information Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2015**

LEMBAR PENGESAHAN

RANCANG BANGUN APLIKASI MEDIA INFORMASI NUTRISI PADA MAKANAN ATAU PRODUK MAKANAN MENGGUNAKAN TEKNOLOGI AUGMENTED REALITY BERBASIS ANDROID

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Bidang Studi Interaksi Grafika dan Seni
Program Studi S-1 Jurusan Teknik Informatika
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

EKO ADHI WIYONO

NRP. 5111100002

Disetujui oleh Pembimbing Tugas Akhir:

1. Wijayanti Nurul Khotimah, S.Kom., M.Sc.
NIP: 19860312 201212 2 004 (pembimbing 1)
2. Ridho Rahman Hariadi, S.Kom., M.Sc.
NIP: 19870213 201404 1 001 (pembimbing 2)



**SURABAYA
JUNI, 2015**

LEMBAR PENGESAHAN

RANCANG BANGUN APLIKASI MEDIA INFORMASI NUTRISI PADA MAKANAN ATAU PRODUK MAKANAN MENGGUNAKAN TEKNOLOGI AUGMENTED REALITY BERBASIS ANDROID

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Bidang Studi Interaksi Grafika dan Seni
Program Studi S-1 Jurusan Teknik Informatika
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

EKO ADHI WIYONO
NRP. 5111100002

Disetujui oleh Pembimbing Tugas Akhir:

1. Wijayanti Nurul Khotimah, S.Kom., M.Sc.
NIP: 19860312 201212 2 004 (pembimbing 1)
2. Ridho Rahman Hariadi, S.Kom., M.Sc.
NIP: 19870213 201404 1 001 (pembimbing 2)



**SURABAYA
JUNI, 2015**

RANCANG BANGUN APLIKASI MEDIA INFORMASI NUTRISI PADA MAKANAN ATAU PRODUK MAKANAN MENGGUNAKAN TEKNOLOGI AUGMENTED REALITY BERBASIS ANDROID

Nama : Eko Adhi Wiyono
NRP : 5111100002
Jurusan : Teknik Informatika, FTIf, ITS
Dosen Pembimbing I : Wijayanti Nurul Khotimah, S.Kom.,
M.Sc.
Dosen Pembimbing II : Ridho Rahman Hariadi, S.Kom.,
M.Sc.

ABSTRAK

Obesitas mengalami kenaikan dua kali lipat semenjak tahun 1980 dan menjadi penyebab kematian nomor lima tertinggi pada setiap tahun. WHO (World Health Organization) mencatat bahwa sekitar 2.8 juta orang dewasa pada setiap tahunnya mengalami kematian yang disebabkan oleh obesitas. Hal tersebut dikarenakan manusia mengonsumsi makanan tanpa memperhatikan kebutuhan kalori dan kandungan nutrisi. Dengan mobilitas manusia yang semakin bertambah, sudah banyak dikembangkan aplikasi perangkat bergerak berbasis android yang mendukung untuk memantau kebutuhan kalori dan nutrisi makanan atau produk makanan seperti aplikasi Noom Coach, Calorie Counter, Lose It, dan lain sebagainya. Namun, aplikasi pada perangkat bergerak tersebut belum interaktif karena pengguna harus mencari terlebih dahulu nama makanan yang dikonsumsi kemudian dicari kandungan nutrisinya. Aplikasi pada Tugas Akhir ini menggunakan teknologi augmented reality teks berbasis android untuk menampilkan informasi nutrisi secara menarik dan informatif. Informasi yang ditampilkan berupa kalori, lemak, karbohidrat dan protein per takaran saji. Takaran saji

ditentukan oleh basis data Fatsecret. Melalui aplikasi ini, pengguna bisa mendapatkan informasi nutrisi hanya dengan mengambil gambar dari suatu objek makanan. Dengan menggunakan metode klasifikasi gambar, aplikasi bisa mengidentifikasi makanan atau produk makanan dari foto untuk mendapatkan informasi nama objek. Informasi nama objek tersebut dikirimkan ke webservice yang disediakan oleh FatSecret untuk mendapatkan informasi kandungan nutrisi. Berdasarkan hasil uji coba pada aplikasi Tugas Akhir ini, didapatkan rata-rata kebenaran dalam mengidentifikasi nama dan nutrisi sebesar 92%. Sedangkan rata-rata waktu yang diperlukan untuk identifikasi nama dan nutrisi yaitu 9.295 detik.

Kata kunci: *android, augmented reality, asupan nutrisi, informasi nutrisi, obesitas.*

DESIGN AND IMPLEMENTATION OF FOODS AND FOOD PRODUCTS NUTRITION USING AUGMENTED REALITY TECHNOLOGY IN ANDROID

Name	: Eko Adhi Wiyono
NRP	: 5111100002
Department	: Informatics Engineering, FTIf, ITS
Advisor I	: Wijayanti Nurul Khotimah, S.Kom., M.Sc.
Advisor II	: Ridho Rahman Hariadi, S.Kom., M.Sc.

ABSTRACT

Obesity has increased two-fold since 1980 and became the fifth highest cause of death in each year. WHO (World Health Organization) noted that about 2.8 million adults every year to experience deaths caused by obesity. That is because humans consume food without regard to the needs of calories and nutritional content. With the increasing mobility of people, has been developed android based mobile device applications that support to monitor caloric and nutritional needs food or food products such as application Coach Noom, Calorie Counter, Lose It, and etc. However, applications on mobile devices are not interactive because the user must seek first the name of the food consumed and then look for nutritional content. Applications in this final project using augmented reality technology android-based text to display nutritional information in an interesting and informative. The information displayed in the form of calories, fat, carbohydrate and protein per serving. Serving size is determined by FatSecret database. Through this application, users can get the nutritional information simply by taking a picture of an object food. By using the image classification method, an application can

identify the food or food products from photographs to obtain information object name. Information object name is sent to the web service provided by FatSecret to get nutritional information. Based on trial results on this final project application, obtained an average of truth in identifying the name and nutrition by 92%. While the average time required to identify the name and nutrients is 9.295 seconds.

Keywords: *android, augmented reality, nutrient intake, nutritional information, obesity.*

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Alhamdulillahirabbil'alamin, segala puji syukur kepada Allah SWT atas segala karunia dan rahmat-Nya penulis dapat menyelesaikan tugas akhir yang berjudul:

“Rancang Bangun Aplikasi Media Informasi Nutrisi pada Makanan atau Produk Makanan Menggunakan Teknologi Augmented Reality Berbasis Android”

Penulis ingin menyampaikan rasa hormat dan terima kasih kepada pihak-pihak yang telah membantu penulis dalam penyelesaian tugas akhir ini, terutama kepada:

1. Ayah Panggih Wiyono, ibunda Istianah, dan adik saya Ringga Arisgi Wiyono yang tiada hentinya setiap hari memberikan semangat secara lahir dan bathin. Adakalanya yang dicapai penulis hingga saat ini mungkin bukan karena dikabulkannya doa dari sang penulis, melainkan jawaban dari segala doa keluarga yang terus mengucur tiada henti untuk penulis.
2. Prof.Ir.Supeno Djanali, M.Sc., Ph.D. sebagai dosen wali penulis yang turut memberikan saran dan membimbing penulis dalam menentukan mata kuliah yang akan diambil pada setiap semester pembelajaran di Jurusan Teknik Informatika ITS selama ini.
3. Ibu Wijayanti Nurul Khotimah, S.Kom, M.Sc. dan Bapak Ridho Rahman Hariadi, S.Kom., M.Sc. yang telah bersedia untuk menjadi dosen pembimbing tugas akhir sehingga penulis dapat mengerjakan tugas akhir dengan arahan dan bimbingan yang baik hingga akhir.
4. Teman dan sekaligus keluarga kedua di kampus, Mahasiswa Teknik Informatika Angkatan 2011 yang telah berjuang bersama – sama selama menempuh pendidikan di

Jurusan ini. Setelah orang tua, kalian adalah alasan kedua saya bertahan di kampus tercinta ini.

5. Himpunan Mahasiswa Teknik Computer-Informatika ITS yang telah mempertemukan penulis dengan orang-orang hebat serta memberikan tempat untuk balas budi karena telah menjadikan Mahasiswa Teknik Informatika Angkatan 2011 sebagai keluarga.
6. Mas M. Misbachul Huda dan Ardhiansyah Baskara yang telah memberikan inspirasi dalam pencarian topik dan pengerjaan tugas akhir.
7. Ahmad Hayam Brilian yang telah mengajarkan tentang bagaimana pengolahan dan klasifikasi citra menggunakan bahasa Python.
8. Wilik dan Suliadi Marsetya yang telah mengajarkan tentang bagaimana implementasi client-server antara Android dan Webservice.
9. Muhammad Bagus Andra dan Putu Wiramaswara Widya yang telah mengajarkan tentang bagaimana menggunakan Flask Python sebagai Webservice.
10. Teman-teman di Lab. IGS, Rizaldi Tri Yanuar yang mau berbagi penggunaan komputer di lab, serta Rizka Wakhidatus Sholikhah, M. Chaqiqi Mudafi, Askary Muhammad, Didik Purwanto, M Iqbal Rustamadji sebagai teman seperjuangan pengerjaan tugas akhir di Lab. IGS.
11. Teman-teman satu bimbingan RMK IGS, Maranu Toto Negoro, Risal Andika Tridisaputra, Febry Amin Nur Hidayah, Mahendra Harsa Wardhana, Firda Ainur Ramadhani, Hashfi Alfian Ciyuda yang bersepakat untuk berjuang lulus bersama di satu anak bimbingan Ibu Wijayanti Nurul Khotimah, S.Kom., M.Sc.
12. Para sahabat hura-hura selama kuliah, Astari Larasati, Farras Kinan, Monika Maytri, M. Aprialdi Rizky Pratama, Andina Triya Ramadhayanti, dan Friska Ayu Listya yang selalu menjadi tempat untuk berbagi sedih dan tawa di bangku perkuliahan.

13. Serta pihak–pihak lain yang turut membantu penulis baik secara langsung maupun tidak langsung, yang namanya tidak penulis sebutkan disini.

Penulis telah berusaha sebaik–baiknya dalam menyusun tugas akhir ini, mohon maaf apabila ada kesalahan dan kata-kata yang dapat menyinggung perasaan. Penulis berharap tugas akhir ini dapat menjadi media informasi nutrisi yang informatif dan dapat menjadi solusi dalam pengontrolan asupan gizi sehari–hari.

Surabaya, Juni 2015

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN.....	vii
ABSTRAK	ix
ABSTRACT	xi
KATA PENGANTAR.....	xiii
DAFTAR ISI	xvii
DAFTAR GAMBAR	xxi
DAFTAR TABEL	xxiii
DAFTAR KODE SUMBER	xxv
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Permasalahan.....	3
1.3. Batasan Permasalahan	4
1.4. Tujuan.....	5
1.5. Manfaat.....	5
1.6. Metodologi	5
1.6.1. Penyusunan Proposal Tugas Akhir.....	5
1.6.2. Studi Literatur.....	6
1.6.3. Analisis dan Desain Perangkat Lunak.....	6
1.6.4. Implementasi Perangkat Lunak	6
1.6.5. Pengujian dan Evaluasi	7
1.6.6. Penyusunan Buku Tugas Akhir	7
1.7. Sistematika Penulisan.....	7
BAB II TINJAUAN PUSTAKA	9
2.1. Nutrisi Makanan	9
2.2. Augmented Reality.....	16
2.3. Sistem Operasi Android dan Android Studio	18
2.4. SURF (<i>Speeded-Up Robust Features</i>).....	18
2.4.1. Pendeteksian <i>Keypoint</i>	19
2.4.2. Pendeskripsian <i>Keypoint</i>	20
2.4.3. Pencocokan Deskriptor.....	20
2.5. OpenCV.....	21
2.6. Platform Fatsecret API	22

2.7.	Webservice	23
2.8.	Flask Python	24
BAB III ANALISIS DAN PERANCANGAN SISTEM.....		25
3.1.	Analisis Perangkat Lunak	25
3.1.1.	Deskripsi Umum Perangkat Lunak.....	25
3.1.2.	Spesifikasi Kebutuhan Perangkat Lunak	26
3.1.3.	Identifikasi Pengguna	27
3.2.	Perancangan Perangkat Lunak.....	27
3.2.1.	Model Kasus Penggunaan	28
3.2.2.	Definisi Aktor	28
3.2.3.	Definisi Kasus Penggunaan	29
3.2.4.	Arsitektur Umum Sistem	33
3.2.5.	Rancangan Antarmuka Aplikasi	34
3.2.6.	Rancangan Proses Penggunaan Aplikasi	40
3.2.7.	Rancangan Proses Klasifikasi Gambar	42
BAB IV IMPLEMENTASI		45
4.1.	Lingkungan Pembangunan	45
4.1.1.	Lingkungan Pembangunan Perangkat Keras	45
4.1.2.	Lingkungan Pembangunan Perangkat Lunak	45
4.2.	Implementasi Antarmuka Aplikasi Android.....	46
4.2.1.	Implementasi Antarmuka Halaman Utama	46
4.2.2.	Implementasi Antarmuka Pengiriman Gambar ke <i>Webservice</i>	47
4.2.3.	Implementasi Antarmuka Rangkuman Nutrisi	48
4.2.4.	Implementasi Antarmuka <i>Server Connection</i>	49
4.2.5.	Implementasi Antarmuka Masukkan Alamat IP <i>Server</i>	50
4.3.	Implementasi Antarmuka <i>Webservice</i>	51
4.3.1.	Implementasi Lapisan <i>Server</i> pada <i>Webservice</i> ..	51
BAB V PENGUJIAN DAN EVALUASI		57
5.1.	Lingkungan Pembangunan	57
5.2.	Skenario Pengujian	57
5.2.1.	Pengujian Skenario 1 dan Evaluasi	59
5.2.2.	Pengujian Skenario 2 dan Evaluasi	61
5.2.3.	Pengujian Skenario 3 dan Evaluasi	63

5.2.4.	Pengujian Skenario 4 dan Evaluasi	65
5.2.5.	Pengujian Skenario 5 dan Evaluasi	68
5.2.6.	Pengujian Skenario 6 dan Evaluasi	72
5.2.7.	Pengujian Skenario 7 dan Evaluasi	75
5.2.8.	Pengujian Skenario 8 dan Evaluasi	77
5.2.9.	Pengujian Skenario 9 dan Evaluasi	81
5.2.10.	Pengujian Skenario 10 dan Evaluasi	83
5.2.11.	Pengujian Skenario 11 dan Evaluasi	86
5.2.12.	Pengujian Skenario 12 dan Evaluasi	88
5.2.13.	Pengujian Skenario 13 dan Evaluasi	91
5.2.14.	Pengujian Skenario 14 dan Evaluasi	93
BAB VI KESIMPULAN DAN SARAN.....		97
6.1.	Kesimpulan.....	97
6.2.	Saran.....	98
DAFTAR PUSTAKA		99
LAMPIRAN (A) KODE SUMBER		101
BIODATA PENULIS		115

DAFTAR TABEL

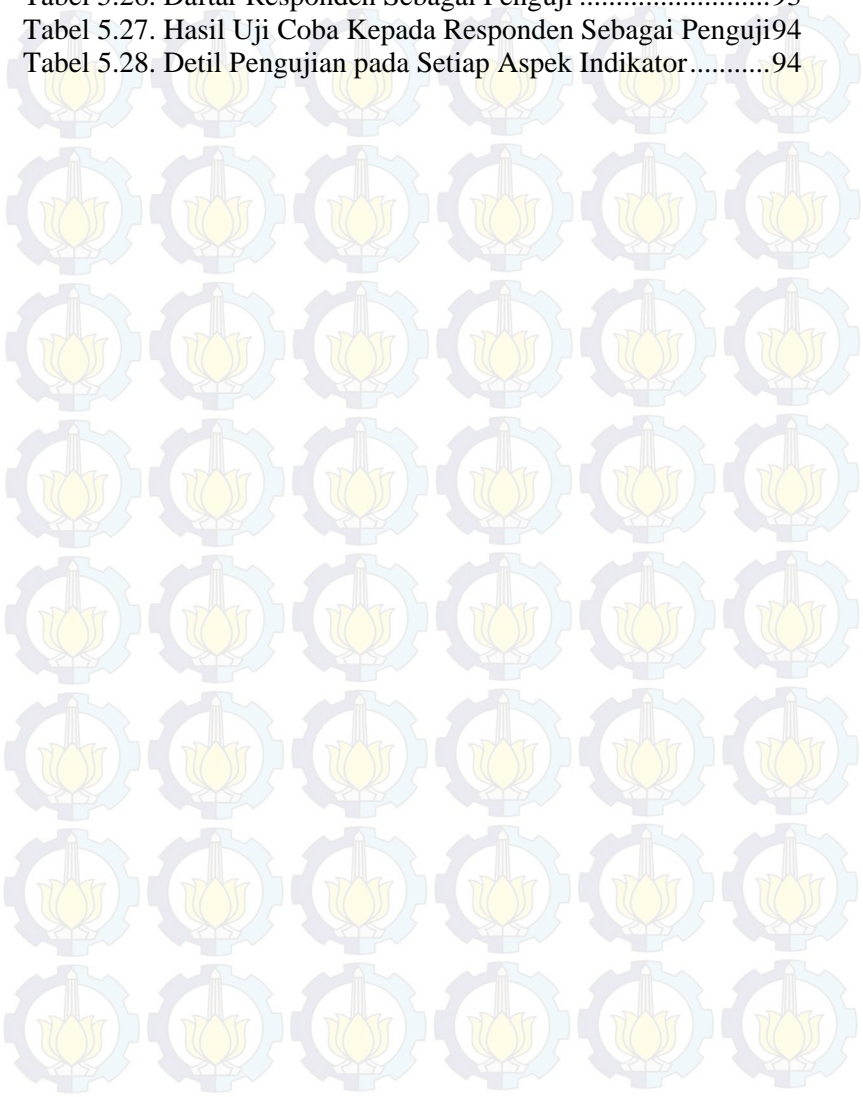
Tabel 2.1. Daftar Nutrisi Setiap Takaran Per 100gr	9
Tabel 3.1. Definisi Aktor.....	28
Tabel 3.2. Definisi Kasus Penggunaan.....	29
Tabel 3.3. Spesifikasi Kasus Penggunaan Mengambil Foto Objek Makanan atau Produk Makanan	30
Tabel 3.4. Spesifikasi Kasus Penggunaan Mendapatkan Informasi Nutrisi dalam Bentuk <i>Augmented Reality</i> Teks.....	31
Tabel 3.5. Spesifikasi Kasus Penggunaan Mengidentifikasi Foto Objek Makanan atau Produk Makanan.	32
Tabel 5.1. Skenario Pengujian 1	59
Tabel 5.2. Skenario Pengujian 2.....	61
Tabel 5.3. Skenario Pengujian 3.....	63
Tabel 5.4. Dokumentasi Hasil Pengujian 3	64
Tabel 5.5. Skenario Pengujian 4.....	66
Tabel 5.6. Dokumentasi Hasil Pengujian 4	67
Tabel 5.7. Skenario Pengujian 5.....	69
Tabel 5.8. Dokumentasi Hasil Pengujian 5	70
Tabel 5.9. Skenario Pengujian 6.....	72
Tabel 5.10. Dokumentasi Hasil Pengujian 6	73
Tabel 5.11. Skenario Pengujian 7	75
Tabel 5.12. Dokumentasi Hasil Pengujian 7	76
Tabel 5.13. Skenario Pengujian 8.....	77
Tabel 5.14. Dokumentasi Hasil Pengujian 8	78
Tabel 5.15. Skenario Pengujian 9.....	81
Tabel 5.16. Dokumentasi Hasil Pengujian 9	82
Tabel 5.17. Skenario Pengujian 10.....	83
Tabel 5.18. Dokumentasi Hasil Pengujian 10	84
Tabel 5.19. Skenario Pengujian 11	86
Tabel 5.20. Dokumentasi Hasil Pengujian 11	87
Tabel 5.21. Skenario Pengujian 12.....	88
Tabel 5.22. Dokumentasi Hasil Pengujian 12	89
Tabel 5.23. Hasil Pengujian 13	91
Tabel 5.24. Hasil Kalkulasi Prosentase Kebenaran.....	92

Tabel 5.25. Hasil Kalkulasi Waktu Identifikasi92

Tabel 5.26. Daftar Responden Sebagai Penguji93

Tabel 5.27. Hasil Uji Coba Kepada Responden Sebagai Penguji94

Tabel 5.28. Detil Pengujian pada Setiap Aspek Indikator 94



DAFTAR GAMBAR

Gambar 2.1. <i>Augmented Reality</i> dengan Objek Buah Apel	17
Gambar 2.2. Pendeteksian <i>Keypoint Point</i>	19
Gambar 2.3. Pendeskripsian <i>Keypoint</i>	20
Gambar 2.4. Visualisasi Pencocokan Deskriptor Gambar	21
Gambar 2.5. Contoh Penggunaan <i>Fatsecret</i> untuk Pencarian Nutrisi Buah Apel.....	23
Gambar 3.1. Model Kasus Penggunaan Aplikasi	28
Gambar 3.2. Desain Arsitektur Aplikasi	34
Gambar 3.3. Rancangan Antarmuka Halaman Utama	35
Gambar 3.4. Rancangan Antarmuka Pengiriman Gambar ke Server	36
Gambar 3.5. Rancangan Antarmuka Rangkuman Nutrisi	37
Gambar 3.6. Rancangan Antarmuka <i>Server Connection</i>	38
Gambar 3.7. Rancangan Antarmuka Masukkan Alamat IP <i>Server</i>	39
Gambar 3.8. Rancangan Antarmuka <i>Webservice</i>	40
Gambar 3.9. Diagram Alir Proses Penggunaan Aplikasi	41
Gambar 3.10. Diagram Alir Proses Klasifikasi Gambar	42
Gambar 3.11. Proses Ekstraksi Fitur <i>Keypoint</i> Gambar	43
Gambar 3.12. Proses Perhitungan Kemiripan Gambar	44
Gambar 4.1. Antarmuka Halaman Utama	47
Gambar 4.2. Antarmuka Pengiriman Gambar ke <i>Webservice</i>	48
Gambar 4.3. Antarmuka Rangkuman Nutrisi.....	49
Gambar 4.4. Antarmuka <i>Server Connection</i>	50
Gambar 4.5. Antarmuka Masukkan Alamat IP <i>Server</i>	51
Gambar 5.1. Modus Pengambilan Gambar Objek Makanan atau Produk Makanan	60
Gambar 5.2. Pengujian Tombol <i>Identify Object</i>	62
Gambar 5.3. Pengujian Tombol <i>Connection</i>	62
Gambar 5.4. Contoh Uji Coba Pengujian Aplikasi pada Buah Pisang	65
Gambar 5.5. Identifikasi Kemiripan Buah Jeruk dan Pisang	67

Gambar 5.6. Contoh Uji Coba Pengujian Aplikasi pada Buah Jeruk	68
Gambar 5.7. Identifikasi Kemiripan Buah Apel dan Jeruk	70
Gambar 5.8. Identifikasi Kemiripan Buah Apel dan Pisang	71
Gambar 5.9. Contoh Uji Coba Pengujian Aplikasi pada Buah Apel	72
Gambar 5.10. Contoh Uji Coba Pengujian Aplikasi pada Produk Indomie.....	74
Gambar 5.11. Contoh Uji Coba Pengujian Aplikasi pada Produk Pringles.....	77
Gambar 5.12. Contoh Uji Coba Pengujian Aplikasi pada Buah Lemon.....	79
Gambar 5.13. Identifikasi Kemiripan Buah Lemon dan Pisang ..	80
Gambar 5.14. Identifikasi Kemiripan Buah Lemon dan Apel.....	80
Gambar 5.15. Contoh Uji Coba Pengujian Aplikasi pada Produk Pocky	83
Gambar 5.16. Contoh Uji Coba Pengujian Aplikasi pada Produk Oreo	85
Gambar 5.17. Contoh Uji Coba Pengujian Aplikasi pada Produk Snickers	88
Gambar 5.18. Contoh Uji Coba Pengujian Aplikasi pada Produk Tango.....	90

DAFTAR KODE SUMBER

Kode Sumber 2.1. Contoh Kode Program untuk Mengakses Basis Data Nutrisi <i>Fatsecret</i>	22
Kode Sumber 2.2. Contoh Penggunaan Flask Python.....	24
Kode Sumber 4.1. Implementasi <i>Server</i> pada <i>Webservice</i>	52
Kode Sumber 4.2. Implementasi Ekstraksi Fitur <i>Keypoint</i>	53
Kode Sumber 4.3. Implementasi Penyimpanan Hasil Ekstraksi Fitur	54
Kode Sumber 4.4. Implementasi Pembacaan Hasil Ekstraksi Fitur Gambar dari Android Untuk Dikomputasi	55
Kode Sumber 4.5. Implementasi Pencocokan Gambar di Basis data Direktori	56
Kode Sumber 4.6. Implementasi Mendapatkan Rangkuman Nutrisi Objek Makanan	56
Kode Sumber 7.1. Kode <i>MainActivity.java</i> (Bagian 1).....	102
Kode Sumber 7.2. Kode <i>MainActivity.java</i> (Bagian 2).....	103
Kode Sumber 7.3. Kode <i>MainActivity.java</i> (Bagian 3).....	104
Kode Sumber 7.4. Kode <i>MainActivity.java</i> (Bagian 4).....	107
Kode Sumber 7.5. Kode <i>MainActivity.java</i> (Bagian 5).....	108
Kode Sumber 7.6. Kode <i>CameraView.java</i>	109
Kode Sumber 7.7. Kode <i>Activity_main.xml</i>	113

BIODATA PENULIS



Penulis lahir di Jember, 29 April 1992. Penulis menempuh pendidikan di SD Negeri 3 Menampu (1999-2005), SMP Negeri 1 Puger (2005-2008), dan SMA Negeri 1 Jember (2008-2011). Penulis melanjutkan pendidikan sarjana di Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember Surabaya. Dalam menyelesaikan pendidikan S1, penulis mengambil bidang minat Interaksi Grafika dan Seni (IGS). Selain aktif sebagai mahasiswa di bidang akademik, penulis juga aktif dalam berbagai kegiatan nonakademik. Tahun ke-2 sebagai mahasiswa, penulis aktif sebagai Staff Departemen Pengembangan Sumber Daya Mahasiswa dan sebagai Organizer Committee kaderisasi Kabinet HMTC Bersatu. Penulis aktif dalam kepanitiaan National Seminar Technology - SCHEMATICS 2012 ITS dan juga aktif di bidang sosial sebagai Volunteer TC Mengabdi di Desa Tempurejo, Surabaya. Tahun ketiga sebagai mahasiswa, penulis aktif sebagai Staff Ahli Departemen Sumber Daya Mahasiswa serta sebagai Instructor Committee kaderisasi Kabinet HMTC Bersahabat dan GERIGI ITS 2013. Penulis juga aktif sebagai panitia National Seminar Technology - SCHEMATICS 2013 ITS. Penulis memiliki pengalaman kerja praktik selama satu bulan di Biro Sistem Informasi PT. Wijaya Karya (Persero) Tbk, Jakarta Timur sebagai analis sistem untuk penambahan kebutuhan aplikasi WIKA ScoreCard. Penulis dapat dihubungi melalui alamat email ekoadhiwiyono@gmail.com

BAB I

PENDAHULUAN

Pada bab ini dipaparkan mengenai garis besar tugas akhir yang meliputi latar belakang, tujuan, rumusan dan batasan permasalahan, metodologi pembuatan tugas akhir, dan sistematika penulisan.

1.1. Latar Belakang

Obesitas sudah mengalami kenaikan dua kali lipat semenjak tahun 1980 dan menjadi penyebab kematian nomor lima tertinggi pada setiap tahun. Sekitar 2.8 juta orang dewasa pada setiap tahunnya mengalami kematian yang disebabkan oleh obesitas. Badan Organisasi Kesehatan Dunia WHO (*World Health Organization*) mendefinisikan bahwa seseorang mengalami kelebihan berat badan apabila indeks massa berat lebih dari 25 kg/m² dan menjadi obesitas apabila indeks massa berat melebihi 30 kg/m². Dalam rangka memerangi obesitas, WHO (*World Health Organization*) merekomendasikan bahwa setiap individu harus mengurangi kalori dari lemak dan gula berlebih, serta menjaga pola hidup sehat dengan berolahraga secara teratur minimal 150 menit per minggu [1]. Oleh karena itu, setiap individu disarankan untuk memperhatikan kandungan nutrisi pada makanan yang dikonsumsi sehingga bisa mengurangi kalori dari lemak dan gula darah berlebih.

Nutrisi adalah bidang ilmu yang mempelajari bagaimana makanan dapat mempengaruhi kesehatan dan kelangsungan hidup manusia. Manusia memerlukan makanan untuk tumbuh, bereproduksi, dan memelihara kesehatan yang baik. Tanpa makanan, tubuh manusia tidak dapat menjaga suhunya, membangun atau memperbaiki jaringan, atau memelihara detak jantung. Memakan makanan yang benar dapat menghindarkan manusia dari berbagai penyakit atau sembuh lebih cepat ketika

penyakit menyerang. Faktor dan fungsi penting lain dipenuhi dengan zat-zat kimia di dalam makanan manusia yang disebut nutrisi. Nutrisi dikategorikan sebagai karbohidrat, protein, lemak, vitamin-vitamin dan mineral [2]. Berat badan dapat diturunkan dengan menggunakan cara membatasi asupan nutrisi. Faktor pengali untuk energi yang umum diterima oleh banyak orang adalah sebagai berikut: 1 gram karbohidrat menghasilkan 4 kkal, 1 gram protein 4 kkal, dan 1 gram lemak kkal, sedangkan alkohol 7 kkal [3].

Kalori merupakan satuan tenaga yang akan dibakar oleh tubuh. Banyaknya jumlah kalori yang dibutuhkan dapat berbeda-beda pada setiap orang. Kebutuhan akan energi yang berbeda dipengaruhi oleh berat badan, jenis kelamin, umur dan aktivitas fisik yang dilakukan. Menentukan kebutuhan kalori dan asupan nutrisi tubuh per hari merupakan suatu kendala bagi kita. Kendala yang sering dihadapi adalah banyak sekali produk-produk makanan yang belum diketahui total kalori dan nutrisi yang terkandung di dalamnya. Dengan mobilitas manusia yang semakin bertambah, sudah banyak dikembangkan perangkat bergerak berbasis android yang mendukung untuk memantau kebutuhan kalori dan nutrisi makanan atau produk makanan. Misalnya saja aplikasi *Nutrition Facts* [4], aplikasi ini memberikan informasi berupa basis data nutrisi dan kalori yang tersimpan dalam suatu makanan atau produk makanan. Pengguna aplikasi ini dihadapkan pada fitur *search* untuk mencari kandungan nutrisi berdasarkan makanan atau produk makanan yang dipilih. Informasi nutrisi makanan yang dipilih berupa kandungan air, kalori, protein, total lemak, dan karbohidrat. Namun kekurangan dari aplikasi ini adalah pengguna tidak bisa secara langsung mendapatkan informasi kandungan nutrisi. Pengguna harus mencari terlebih dahulu makanan atau produk makanan apa yang akan dikonsumsi. Pengguna juga terbatas pada basis data makanan atau produk makanan aplikasi tersebut saja.

Perkembangan *platform android* sudah menggunakan teknologi *augmented reality* untuk mendukung kinerja aplikasi.

Perkembangan *augmented reality* semakin hari semakin pesat. Dari aplikasi yang sederhana hingga aplikasi yang kompleks. Misalnya saja aplikasi *augmented reality* untuk pemasaran produk, rehabilitasi pasca sakit *stroke*, hingga *game 3D* yang dapat berinteraksi dengan pengguna. Teknologi *augmented reality* dirasa masih baru dan menarik bagi masyarakat. Beberapa riset menemukan bahwa implementasi dari *augmented reality* pada aplikasi perangkat bergerak bisa merepresentasikan informasi secara visual [5]. Oleh karena itu, maka pada Tugas Akhir ini diimplementasikan *augmented reality* untuk merepresentasikan informasi nutrisi secara menarik.

Rancang bangun media informasi nutrisi pada makanan atau produk makanan pada tugas akhir ini, menggunakan teknologi *augmented reality* untuk menampilkan informasi nutrisi. Informasi yang ditampilkan berupa kandungan nutrisi yaitu kalori, lemak, karbohidrat dan protein. Misalnya pengguna aplikasi ini akan mengonsumsi 1 buah apel, 2 lembar roti tawar, dan satu gelas susu. Untuk mengetahui kandungan nutrisi di dalam makanan atau produk makanan tersebut, pengguna bisa mendapatkan informasi dengan mengambil foto makanan atau produk makanan. Aplikasi akan menampilkan informasi nutrisi berupa kalori, lemak, karbohidrat dan protein objek yang difoto dalam bentuk *augmented reality*.

Aplikasi ini dibangun dengan harapan pengguna perangkat bergerak bisa mengontrol makanan atau produk makanan apa saja yang akan dimakan dengan memperhatikan kandungan nutrisi di dalamnya kapanpun dan dimanapun. Dengan teknologi *augmented reality* yang masih baru di khalayak umum diharapkan bisa menarik perhatian pengguna untuk memperhatikan lebih kalori, karbohidrat, protein dan lemak pada makanan atau produk makanan yang akan dikonsumsi sesuai dengan kebutuhan nutrisi di setiap harinya.

1.2. Rumusan Permasalahan

Rumusan masalah yang diangkat dalam tugas akhir ini adalah sebagai berikut:

1. Bagaimana mengidentifikasi jenis makanan atau produk makanan dari masukan berupa gambar serta mendapatkan informasi nutrisi berupa kalori, karbohidrat, protein dan lemak yang terkandung?
2. Bagaimana menampilkan informasi nutrisi berupa kalori, karbohidrat, protein dan lemak terkandung di dalam makanan kepada pengguna perangkat bergerak berbasis Android yang menarik menggunakan teknologi *augmented reality* teks?

1.3. Batasan Permasalahan

Permasalahan yang dibahas dalam tugas akhir ini memiliki beberapa batasan, di antaranya sebagai berikut:

1. Aplikasi dibangun khusus untuk perangkat bergerak dengan sistem operasi Android dengan versi minimum 4.0 yang terhubung dengan internet.
2. Aplikasi hanya bisa mengidentifikasi nutrisi suatu makanan atau produk makanan yang sudah dilatih sebelumnya.
3. Aplikasi menggunakan informasi nutrisi yang terkandung dalam makanan atau produk makanan menggunakan basis data *FatSecret* (www.fatsecret.com).
4. Aplikasi hanya dapat dijalankan pada jaringan lokal yang terhubung antara aplikasi dan *webservice*.
5. Aplikasi hanya dapat berkomunikasi dengan *webservice* apabila *Windows Firewall* pada *server* dalam kondisi tidak aktif.
6. Jarak antara kamera dan objek makanan dalam pengambilan gambar maksimal 15-20cm, serta tidak memiliki latar belakang objek yang terlalu ramai.
7. *Augmented relaity* teks berupa kandungan nutrisi yang hanya bisa muncul di layar modus kamera.

1.4. Tujuan

Tujuan dalam pembuatan tugas akhir ini antara lain adalah sebagai berikut:

1. Membuat aplikasi yang bisa mengidentifikasi jenis makanan atau produk makanan dari masukan berupa gambar serta mendapatkan informasi nutrisi berupa kalori, karbohidrat, protein dan lemak yang terkandung.
2. Menampilkan informasi nutrisi berupa kalori, karbohidrat, protein dan lemak terkandung di dalam makanan atau produk makanan kepada pengguna perangkat bergerak berbasis Android menggunakan teknologi *augmented reality* teks sehingga lebih menarik.

1.5. Manfaat

Dengan dibangunnya perangkat lunak media informasi nutrisi pada makanan atau produk makanan menggunakan teknologi *augmented reality* teks berbasis Android diharapkan kita bisa lebih memerhatikan kandungan nutrisi makanan yang akan dikonsumsi terutama kalori, karbohidrat, lemak dan protein sesuai dengan kebutuhan setiap hari. Dikemas dengan teknologi *augmented reality* teks diharapkan bisa lebih menarik pengguna aplikasi serta bisa menjadi media informasi nutrisi yang informatif.

1.6. Metodologi

Tahapan–tahapan yang dilakukan dalam pengerjaan tugas akhir ini adalah sebagai berikut.

1.6.1. Penyusunan Proposal Tugas Akhir.

Tahap awal pengerjaan tugas akhir adalah penyusunan proposal. Proposal ini mengajukan gagasan mengenai “Rancang Bangun Aplikasi Media Informasi Nutrisi pada Makanan atau Produk Makanan Menggunakan Teknologi *Augmented Reality* Berbasis Android”. Pada tahap ini dilakukan pengumpulan dan penggalan informasi dan studi literatur yang diperlukan dalam proses perancangan dan implementasi sistem yang akan dibangun.

1.6.2. Studi Literatur

Studi literatur yang dilakukan dalam pengerjaan tugas akhir ini adalah dengan mengumpulkan informasi yang diperlukan yaitu berupa:

- i. Penggunaan metode klasifikasi gambar untuk mengenali objek dengan mengambil foto objek;
- ii. Pengembangan *webservice* untuk mengidentifikasi foto objek dan mengirimkan informasi objek berupa nama objek;
- iii. Penggunaan *Fatsecret* API sebagai basis data penyimpanan informasi nutrisi makanan atau produk makanan;
- iv. Penerapan *augmented reality* teks untuk menampilkan informasi nutrisi berupa kalori, karbohidrat, protein dan lemak;
- v. Studi literatur dapat didapatkan melalui buku tutorial, artikel ilmiah, maupun *website*.

1.6.3. Analisis dan Desain Perangkat Lunak

Analisis dan desain perangkat lunak dilakukan bergantung dari hasil analisa kebutuhan sistem. Aplikasi media informasi nutrisi pada makanan atau produk makanan menggunakan teknologi *augmented reality* berbasis android. Aplikasi mampu mengidentifikasi objek untuk mendapatkan informasi nutrisi dari objek yang difoto. Kemudian informasi nutrisi ditampilkan ke layar perangkat bergerak pengguna menggunakan *augmented reality* teks.

1.6.4. Implementasi Perangkat Lunak

Pada tahap implementasi dilakukan pembuatan perangkat lunak yang merupakan implementasi dari rancangan yang telah dibuat. Pada tahap ini dilakukan implementasi klasifikasi gambar untuk mengenali objek dengan mengambil foto objek. Dilanjutkan dengan implementasi *augmented reality* teks untuk menampilkan informasi nutrisi berupa kalori, karbohidrat, lemak dan protein.

1.6.5. Pengujian dan Evaluasi

Tahap ini dilakukan pengujian terhadap sistem menggunakan data atau kasus yang telah disiapkan. Uji coba dilakukan untuk menguji fungsionalitas sistem, mencari masalah yang mungkin timbul pada sistem dan melakukan perbaikan apabila terdapat kekurangan. Uji coba ini dilakukan ke mahasiswa dan masyarakat sekitar ITS kecuali dosen dan staf tata usaha yang aktif sebagai konsumen makanan dan produk makanan saja. Uji coba yang dilakukan melibatkan remaja umur 13 – 21 tahun dan dewasa umur 21 – 40 tahun yang memiliki perangkat bergerak berbasis *android*.

1.6.6. Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat.

1.7. Sistematika Penulisan

Buku tugas akhir ini disusun dengan sistematika penulisan sebagai berikut.

BAB I. PENDAHULUAN

Bab ini berisi mengenai latar belakang, tujuan, dan manfaat dari pembuatan tugas akhir. Selain itu, permasalahan, batasan masalah, metodologi yang digunakan, dan sistematika penulisan juga merupakan bagian dari bab ini.

BAB II. TINJAUAN PUSTAKA

Bab ini berisi penjelasan secara detail mengenai dasar–dasar penunjang dan teori–teori yang digunakan untuk mendukung pembuatan tugas akhir ini.

BAB III. ANALISIS DAN PERANCANGAN SISTEM

Bab ini membahas tahap analisis permasalahan dan perancangan dari sistem yang akan dibangun. Analisis permasalahan membahas permasalahan yang diangkat dalam pengerjaan tugas akhir.

BAB IV. IMPLEMENTASI

Bab ini membahas implementasi dari desain yang telah dibuat pada bab sebelumnya. Bab ini berisi proses implementasi hasil analisis pada semua modul.

BAB V. PENGUJIAN DAN EVALUASI

Bab ini menjelaskan kemampuan perangkat lunak dengan melakukan pengujian kebenaran dan pengujian kinerja dari sistem yang telah dibuat.

BAB VI. KESIMPULAN DAN SARAN

Bab ini merupakan bab terakhir yang menyampaikan kesimpulan dari hasil uji coba yang dilakukan dan saran untuk pengembangan perangkat lunak ke depannya.

BAB II

TINJAUAN PUSTAKA

Pada bab ini berisi penjelasan teori–teori yang berkaitan dengan metode yang diajukan pada pengimplementasian perangkat lunak. Penjelasan ini bertujuan untuk memberikan gambaran secara umum terhadap sistem yang dibuat dan berguna sebagai penunjang dalam pengembangan perangkat lunak.

2.1. Nutrisi Makanan

Nutrisi adalah zat dalam makanan yang dibutuhkan organisme untuk dapat tumbuh dan berkembang dengan baik sesuai dengan fungsinya. Nutrisi dalam jumlah yang memadai dan sesuai dengan kebutuhan akan memberikan energi bagi tubuh untuk dapat tumbuh dan berkembang serta memperbaiki jaringan yang rusak. Kekurangan nutrisi akan membuat tubuh organisme tidak tumbuh dan berkembang sesuai dengan takdirnya, bahkan dapat menyebabkan penyakit hingga berakhir dengan kematian. Terganggunya proses metabolisme tubuh merupakan gejala awal kekurangan nutrisi [6]. Nutrisi berimbang terdiri dari 30% lemak 30% protein 40% karbohidrat. Fungsi Nutrisi adalah sebagai sumber energi, pendukung dan pengatur proses metabolisme, menjaga keseimbangan metabolisme, pembentuk sel-sel jaringan tubuh, memperbaiki sel-sel yang rusak, mempertahankan fungsi organ tubuh, dan lain sebagainya [6]. Daftar nutrisi setiap takaran per 100gr dapat dilihat pada Tabel 2.1.

Tabel 2.1. Daftar Nutrisi Setiap Takaran Per 100gr

No.	Makanan	Protein	Karbohidrat	Lemak	Kalori
1	Beras Merah	3 gr	23 gr	1 gr	111
2	Beras Putih	2 gr	21 gr	0	97

No.	Makanan	Protein	Karbohidrat	Lemak	Kalori
3	Kentang rebus	2 gr	20 gr	0	86
4	Kentang goreng (French Fries)	4 gr	38 gr	17 gr	319
5	Ubi rebus (sweet potato)	1 gr	18 gr	0	76
6	Ubi panggang	2 gr	21 gr	0	90
7	Jagung rebus	3 gr	25 gr	1 gr	108
8	Snack Jagung (Corn Flake)	2 gr	24 gr	0	101
9	Popcorn bumbu karamel	4 gr	79 gr	13 gr	431
10	Mi Instant	10 gr	53 gr	17 gr	400
11	Spaghetti	6 gr	31 gr	1 gr	157
12	Alpukat	2 gr	9 gr	15 gr	160
13	Anggur	1 gr	8 gr	0	32
14	Apel dengan kulit	0	14 gr	0	52
15	Apel tanpa kulit	0	13 gr	0	48
16	Blewah	1 gr	8 gr	0	34
17	Blueberry	1 gr	12 gr	0	57
18	Buah Naga	1 gr	14 gr	1 gr	50
19	Durian	1gr	27 gr	5 gr	147
20	Jambu biji	3 gr	14 gr	1 gr	68
21	Jeruk	1 gr	12 gr	0	47
22	Jus Jeruk	1 gr	10 gr	0	45
23	Kelapa (daging kelapa)	3 gr	15 gr	33 gr	354
24	Kelapa (air kelapa)	1 gr	4 gr	0	19
25	Kelapa (santan mentah)	2 gr	6 gr	24 gr	230
26	Kesemek	1 gr	19 gr	0	70
27	Kismis	3 gr	79 gr	0	299

No.	Makanan	Protein	Karbohidrat	Lemak	Kalori
28	Kiwi	1 gr	15 gr	1 gr	61
29	Leci	1 gr	17 gr	0	66
30	Lemon	1 gr	9 gr	0	29
31	Mangga	1 gr	17 gr	0	65
32	Markisa	2 gr	23 gr	1 gr	97
33	Melon	1 gr	9 gr	0	36
34	Mentimun (dengan kulit)	1 gr	4 gr	0	15
35	Mentimun acar	0	2 gr	0	11
36	Nanas	1 gr	13 gr	0	50
37	Nangka	1 gr	24 gr	0	94
38	Pear	0	11 gr	0	42
39	Pepaya	1 gr	10 gr	0	39
40	Persik (peach)	1 gr	10 gr	0	39
41	Pisang	1 gr	23 gr	0	89
42	Keripik Pisang	2 gr	58 gr	34 gr	519
43	Sawo	0	20 gr	1 gr	83
44	Semangka	1 gr	8 gr	0	30
45	Strawberi	1 gr	8 gr	0	32
46	Srikaya	1 gr	17 gr	0	66
47	Asparagus matang tanpa garam	2 gr	4 gr	0	22
48	Banguang	1 gr	9 gr	0	38
49	Brokoli rebus	1 gr	4 gr	1 gr	22
50	Buncis rebus	2 gr	8 gr	0	35
51	Bunga kol	2 gr	4 gr	0	23
52	Cabai hijau pedas mentah	2 gr	9 gr	0	40
53	Cabai merah pedas mentah	2 gr	9 gr	0	40

No.	Makanan	Protein	Karbohidrat	Lemak	Kalori
54	Jamur kuping kering	9 gr	73 gr	1 gr	284
55	Kedele rebus	17 gr	10 gr	1 gr	173
56	Kubis mentah	1 gr	6 gr	0	25
57	Labu matang tanpa garam	1 gr	5 gr	0	20
58	Pare rebus	1 gr	4 gr	0	19
59	Peterseli (parsley)	3 gr	6 gr	1 gr	36
60	Tauge rebus	2 gr	4 gr	0	21
61	Rebung rebus dengan garam	2 gr	2 gr	0	11
62	Rebung rebus tanpa garam	2 gr	2 gr	0	12
63	Sayur asin kalengan	1 gr	5 gr	0	19
64	Seledri matang tanpa garam	1 gr	4 gr	0	18
65	Selada (home prepared)	1 gr	12 gr	3 gr	78
66	Terong matang tanpa garam	1 gr	9 gr	0	35
67	Terong matang dengan garam	1 gr	8 gr	0	33
68	Tomat Merah mentah	1 gr	4 gr	0	18
69	Tomat Kuning mentah	1 gr	3 gr	0	15
70	Wortel mentah	1 gr	10 gr	0	41
71	Jus wortel	1 gr	9 gr	0	40
72	Cod Atlantik	23 gr	0	1 gr	105
73	Cod Pasifik	23 gr	0	1 gr	105
74	Cumi-cumi (goreng)	18 gr	8 gr	7 gr	175
75	Gurame	23 gr	0	7 gr	162
76	Gurita	30 gr	4 gr	2 gr	164
77	Herring Atlantik	33 gr	0	17 gr	290

No.	Makanan	Protein	Karbohidrat	Lemak	Kalori
78	Herring Pasifik	21 gr	0	18 gr	250
79	Ikan asin mackerel	18 gr	0	25 gr	305
80	Kakap	26 gr	0	2 gr	128
81	Keong ukuran sedang (jenis whelk)	48 gr	16 gr	1 gr	275
82	Keong besar jenis conch (panggang)	26 gr	2 gr	1 gr	130
83	Kepah	24 gr	7 gr	4 gr	172
84	Kepiting (biru)	20 gr	0	2 gr	102
85	Kijing (remis besar)	26 gr	5 gr	2 gr	148
86	Lele	19 gr	0	8 gr	152
87	Lobster	26 gr	3 gr	2 gr	143
88	Mackerel Atlantik	24 gr	0	18 gr	262
89	Mackerel Pasifik	26 gr	0	10 gr	201
90	Remis	23 gr	0	1 gr	112
91	Salmon	25 gr	0	8 gr	182
92	Sarden Atlantik kaleng	25 gr	0	11 gr	208
93	Sarden Pasifik kaleng saus tomat	21 gr	1 gr	10 gr	186
94	Sotong	32 gr	2 gr	1 gr	158
95	Teri goreng (anchovy)	29 gr	0	10 gr	210
96	Tiram jenis abalone (goreng)	20 gr	11 gr	7 gr	189
97	Tiram jenis eastern oyster	14 mg	8 gr	5 gr	137
98	Tiram jenis Pasifik oyster	19 gr	10 gr	5 gr	163
99	Trout (ikan air tawar)	27 gr	0	8 gr	190
100	Tuna	30 gr	0	6 gr	184
101	Udang (shrimp)	21 gr	0	1 gr	99

No.	Makanan	Protein	Karbohidrat	Lemak	Kalori
102	Telur rebus dengan cangkang (hard-boiled)	13 gr	1 gr	11 gr	155
103	Telur rebus tanpa cangkang/ dipecahkan (poached)	13 gr	1 gr	10 gr	142
104	Omelet	11 gr	1 gr	12 gr	157
105	Telur ceplok	14 gr	1 gr	15 gr	196
106	Telur orak arik	11 gr	2 gr	12 gr	167
107	Eggnog	4 gr	14 gr	7 gr	135
108	Putih telur mentah segar	11 gr	1 gr	0	48
109	Kuning telur mentah segar	16 gr	4 gr	27 gr	317
110	Telur mentah segar utuh	13 gr	1 gr	10 gr	143
111	Telur bebek mentah segar utuh	13 gr	1 gr	14 gr	185
112	Telur angsa mentah segar utuh	14 gr	1 gr	13 gr	185
113	Telur puyuh mentah segar utuh	13 gr	0	11 gr	158
114	Telur kalkun mentah segar utuh	14 gr	1 gr	12 gr	171
115	Roti gandum utuh (whole-wheat bread)	13 gr	41 gr	3 gr	247
116	Roti gandum utuh panggang (whole-wheat bread, toasted)	16 gr	41 gr	4 gr	306
117	Roti putih panggang (white bread, toasted)	9 gr	54 gr	4 gr	293
118	Roti bagel telur	11 gr	53 gr	2 gr	278
119	Roti bagel gandum	11 gr	53 gr	1 gr	255
120	Tempe Rebus	18 gr	9 gr	11 gr	196

No.	Makanan	Protein	Karbohidrat	Lemak	Kalori
121	Tofu goreng	17 gr	10 gr	20 gr	271
122	Daging sapi asap (sliced)	19 gr	1 gr	6 gr	139
123	Daging kambing panggang	27 gr	0	3 gr	143
124	Daging kerbau panggang	27 gr	0	2 gr	131
125	Sosis sapi matang	18 gr	0	28 gr	332
126	Corned sapi (kaleng)	19 gr	1 gr	7 gr	142
127	Donat dengan coklat atau gula	5 gr	57 gr	20 gr	417
128	Kue bolu	5 gr	61 gr	3 gr	289
129	Brownies	5 gr	64 gr	16 gr	405
130	Muffin panggang dengan kismis	5 gr	56 gr	9 gr	313
131	Croissant keju	9 gr	47 gr	21 gr	414
132	Waffle panggang	7 gr	49 gr	10 gr	312
133	Biskuit plain komersial	6 gr	48 gr	17 gr	365
134	Coklat wafer	7 gr	72 gr	14 gr	433
135	Cracker meal	9 gr	81	2 gr	383
136	Pie buah (kering)	3 gr	43 gr	16 gr	316
137	Pie apel	2 gr	34 gr	11 gr	237
138	Puding coklat manis	2 gr	23 gr	5 gr	142
139	Kue terang bulan plain	10 gr	71 gr	5 gr	376
140	Kue pastry buah	5 gr	69 gr	11 gr	391
141	Keripik pisang	2 gr	58 gr	34 gr	519
142	Keripik kentang tanpa garam	7 gr	53 gr	35 gr	536
143	Keripik kentang dengan garam	7 gr	50 gr	37 gr	547

No.	Makanan	Protein	Karbohidrat	Lemak	Kalori
144	Cemilan kedelai dengan garam	27 gr	53 gr	7 gr	376
145	Susu bubuk	26 gr	38 gr	27 gr	496
146	Susu kental manis	8 gr	54 gr	9 gr	321
147	Keju mozarella (whole milk)	22 gr	2 gr	22 gr	300
148	Keju Parmesan	38 gr	4 gr	29 gr	431
149	Keju cheddar	25 gr	1 gr	33 gr	403
150	Krim bubuk	5 gr	55	35 gr	545

Sumber: Tabel Nutrisi Dunia Fitnes, 2014

Kalori sesungguhnya sangat dibutuhkan tubuh untuk menghasilkan energi. Namun, jika jumlahnya berlebihan dalam tubuh, ini justru bisa menyebabkan munculnya berbagai penyakit, seperti jantung koroner dan diabetes. Berikut ini adalah cara mudah menghitung kebutuhan kalori harian yang bisa aplikasikan di rumah sebagai acuan dalam menjalankan program diet. Rumus menghitung BBI (Berat Badan Ideal) adalah $BBI = 90\% (\text{tinggi badan} - 100) [7]$.

2.2. Augmented Reality

Realitas tertambah, atau kadang dikenal dengan singkatan bahasa Inggrisnya AR (*Augmented Reality*), adalah teknologi yang menggabungkan benda maya dua dimensi dan ataupun tiga dimensi ke dalam sebuah lingkungan nyata tiga dimensi lalu memproyeksikan benda – benda maya tersebut dalam waktu nyata (real time). Tidak seperti realitas maya (*virtual reality*) yang sepenuhnya menggantikan kenyataan, *augmented reality* sekedar menambahkan atau melengkapi kenyataan [8]. AR dapat diaplikasikan untuk semua indera, tidak hanya visual, termasuk pendengaran, sentuhan, dan penciuman. Gunanya untuk memperkaya pengalaman penggunaannya, membantu persepsi dan

interaksi penggunaanya dengan dunia nyata. Informasi yang ditampilkan oleh benda maya membantu pengguna melaksanakan kegiatan dalam dunia nyata. AR bisa digunakan dalam beberapa bidang penting seperti kesehatan, militer dan industri manufaktur. AR juga telah diaplikasikan dalam perangkat bergerak sudah digunakan oleh orang banyak, seperti telepon genggam [8].

Teknologi *augmented reality* sangat cepat sekali berkembang, di Indonesia sendiri telah banyak aplikasi – aplikasi yang menggunakan teknologi *augmented reality*. *Augmented reality* merupakan terobosan dibidang teknologi yang sangat canggih. Karena dengan teknologi ini kita dapat membuat segala hal yang abstrak atau virtual bisa kelihatan nyata atau real. Teknologi AR sendiri telah dikembangkan dalam berbagai hal, dalam pemanfaatanya teknologi ini dapat digunakan dalam hal *Interactive Games, Presentation, Event, High Tech Environment, Website, Promotion* [8]. Salah satu aplikasi yang memanfaatkan AR dapat dilihat pada Gambar 2.1.



Gambar 2.1. *Augmented Reality* dengan Objek Buah Apel

2.3. Sistem Operasi Android dan Android Studio

Sistem Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat seluler layar sentuh seperti telepon pintar dan komputer tablet. Android awalnya dikembangkan oleh Android, Inc., dengan dukungan finansial dari Google, yang kemudian membelinya pada tahun 2005. Sistem operasi ini dirilis secara resmi pada tahun 2007, bersamaan dengan didirikannya Open Handset Alliance, konsorsium dari perusahaan-perusahaan perangkat keras, perangkat lunak, dan telekomunikasi yang bertujuan untuk memajukan standar terbuka perangkat seluler. Ponsel Android pertama mulai dijual pada bulan Oktober 2008 [9].

Android Studio adalah aplikasi pengembangan baru dan terintegrasi penuh, yang baru saja dirilis oleh Google untuk sistem operasi Android. Android Studio dirancang untuk menjadi peralatan baru dalam pengembangan aplikasi dan juga memberi alternatif lain selain Eclipse yang saat ini menjadi IDE yang paling banyak dipakai. Saat memulai proyek baru dengan Android Studio, struktur proyek akan muncul bersama dengan hampir semua berkas yang ada di dalam direktori SDK, peralihan ke sistem manajemen berbasis Gradle ini memberikan fleksibilitas yang lebih besar pada proses pembangunannya. Android Studio memberikan kesempatan bagi pengguna untuk melihat perubahan visual apapun yang Anda lakukan pada aplikasi secara langsung. Anda juga bisa melihat perbedaannya jika dipasang pada beberapa perangkat Android berbeda, termasuk konfigurasi dan resolusinya secara bersamaan [10].

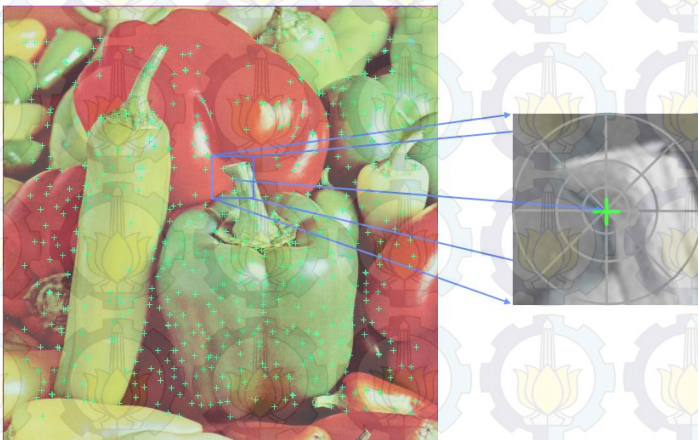
2.4. SURF (*Speeded-Up Robust Features*)

SURF (*Speeded-Up Robust Features*) merupakan sebuah algoritma yang cepat dan akurat untuk proses mendeteksi deskriptor lokal dari kesamaan representasi citra *invariant*. Deskriptor adalah sebuah ciri-ciri dari suatu citra berdasarkan

aturan tertentu dari suatu algoritma. SURF menggunakan citra *integral* untuk meningkatkan kecepatan komputasi. *Integral Image* adalah sebuah citra dimana nilai setiap piksel merupakan akumulasi dari piksel atas dan kirinya. Sebagai contoh, piksel (a,b) memiliki nilai akumulatif untuk semua piksel (x,y) dimana $x \leq a$ dan $y \leq b$. Algoritma ini didasarkan pada kerangka SURF dari hasil disertasi Herbert Bay [11]. Secara umum algoritma ini terdiri dari tiga bagian utama yaitu pendeteksian *keypoint*, pendeskripsian *keypoint*, dan pencocokan deskriptor.

2.4.1. Pendeteksian *Keypoint*

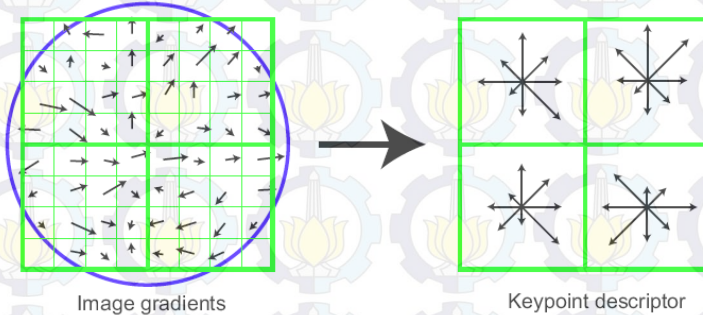
Pendeteksian *keypoint* didasarkan pada matrik Hessian. Tahap ini akan memberikan keluaran berupa *keypoint*. *Keypoint* merupakan *interest point* yang dianggap valid karena sudah berada dibawah batas nilai yang ditetapkan. Determinasi dari matrik Hessian digunakan untuk mendeterminasi lokasi dan skala untuk menyeleksi kandidat *interest point*. Kandidat *interest point* akan dianggap valid apabila mempunyai nilai dibawah batas nilai yang diberikan. Kemudian dilakukan perbaikan pada *box-space* [11]. Pendeteksian *keypoint* dapat dilihat pada Gambar 2.2.



Gambar 2.2. Pendeteksian *Keypoint Point*

2.4.2. Pendeskripsian *Keypoint*

Pendeskripsian *keypoint* dibagi menjadi dua tahap. Tahap pertama adalah menetapkan orientasi berdasarkan informasi dari daerah melingkar di sekitar *interest point* yang terdeteksi.



Gambar 2.3. Pendeskripsian *Keypoint*

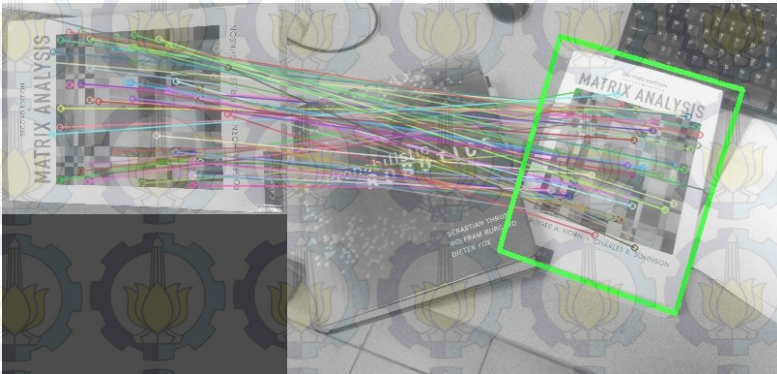
Tahap kedua adalah membuat *grid* sebesar 64 (8x8) yang digunakan untuk menampung deskriptor yang berkorespondensi dengan histogram [11]. Pendeskripsian *keypoint* dapat dilihat pada Gambar 2.3.

2.4.3. Pencocokan Deskriptor

Pencocokan deskriptor adalah pendeskripsian titik-titik fitur menjadi deskriptor vektor agar titik-titik fitur memiliki ketahanan terhadap rotasi, kontras, dan perubahan sudut pandang. Setiap fitur yang terdeteksi akan diberikan orientasi agar tahan terhadap rotasi. Vektor fitur deskriptor yang digunakan pada SURF ini berupa *keypoint* sejumlah 400 dengan *hessian-threshold* sejumlah 100. Kemudian dilakukan perbandingan kecocokan fitur menggunakan metode *Euclidean Distance* antara deskriptor gambar masukan dengan semua deskriptor gambar yang sudah dilatih pada direktori basis data gambar [11]. Perhitungan pencocokan deskriptor dengan metode *Euclidean Distance* dapat dilihat pada Persamaan 2.1

$$D(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_n - y_n)^2} \quad (2.1)$$

Pada Persamaan 2.1. dimana (x, y) adalah fitur deskriptor *keypoint* pada gambar. Sedangkan visualisasi pencocokan deskriptor gambar dapat dilihat pada Gambar 2.4.



Gambar 2.4. Visualisasi Pencocokan Deskriptor Gambar

2.5. OpenCV

OpenCV (*Open Source Computer Vision Library*) adalah sebuah pustaka perangkat lunak yang ditujukan untuk pengolahan citra dinamis secara *real-time*, yang pemngembangannya diawali oleh Intel, dan sekarang didukung oleh Willow Garage dan Itseez. OpenCV dirilis dibawah lisensi permisif BSD yang lebih bebas dari pada GPL, dan memberikan kebebasan sepenuhnya untuk dimanfaatkan secara komersil tanpa perlu mengungkapkan kode sumbernya. Ia juga memiliki antar muka yang mendukung bahasa pemrograman C++, C, Python dan Java, termasuk untuk sistem operasi Windows, Linux, Mac OS, iOS dan Android. OpenCV didisain untuk efisiensi dalam komputasi dan difokuskan pada aplikasi *real-time* [12].

Ditulis dalam bahasa C++, C, Python dan Java yang dioptimalkan, dengan perpustakaan dapat memanfaatkan pengolahan pada prosesor inti majemuk (*multi-core processing*). Mendukung OpenCL, sekaligus memberikan keuntungan dari akselerasi hardware yang dimiliki platform komputasi heterogen. OpenCV telah diadopsi di seluruh dunia, memiliki lebih dari 47 ribu komunitas pengguna dengan estimasi download melebihi 7 juta. Penggunaannya openCV disebutkan, mulai dari seni interaktif, meliputi inspeksi penambangan (*mines inspection*), *stitching maps* di *website*, sampai dengan robotika maju (*advanced*) [12].

2.6. Platform Fatsecret API

Platform *FatSecret* menyediakan informasi nutrisi dari makanan dan produk makanan non lokal yang dapat dipercaya, sehingga pengguna dapat melakukan pemilihan informasi nutrisi makanan apa saja yang akan dimakan. API *Fatsecret* menyediakan akses ratusan basis data nutrisi yang komprehensif bagi pengembang perangkat lunak dengan mengakses informasi gizi melalui REST (*Representational State Transfer*) dan *JavaScript*. API ini gratis untuk digunakan dan didokumentasikan dengan baik secara online beserta contoh kode program. Contoh kode program untuk mengakses basis data nutrisi *Fatsecret* dapat dilihat pada Kode Sumber 2.1.

```
consumer_key = 'consumer key'
consumer_secret = 'consumer secret'

fs = Fatsecret(consumer_key, consumer_secret)
foods = fs.foods_search()
```

Kode Sumber 2.1. Contoh Kode Program untuk Mengakses Basis Data Nutrisi *Fatsecret*

Untuk menggunakan API *Fatsecret*, pengembang harus melakukan registrasi pada *Fatsecret Platform API*. Setelah

registrasi, pengembang akan mendapatkan *REST API Consumer Key* dan *REST API Shared Secret* untuk mengakses API *Fatsecret*. API *Fatsecret* menggunakan OAuth (*Open Authentification*) untuk memverifikasi kebenaran permintaan dan memiliki batas 5000 API panggilan per hari [13]. Basis data *FatSecret* adalah sebagai penyimpanan informasi data nutrisi berupa kalori, karbohidrat, lemak, dan protein. Informasi nama objek yang didapat akan dikirim ke metode *FatSecret Search*, untuk mendapatkan kembali daftar informasi makanan beserta rincian informasi nutrisi.



Gambar 2.5. Contoh Penggunaan *Fatsecret* untuk Pencarian Nutrisi Buah Apel

Masukan dari *Fatsecret Search* berupa tipe data *string* berupa jenis makanan yang akan dicari. Hasil dari *Fatsecret Search* adalah rangkuman nutrisi untuk setiap takaran saji. Salah satu contoh penggunaan *Fatsecret* dapat dilihat pada Gambar 2.5.

2.7. Webservice

Webservice adalah suatu sistem perangkat lunak yang dirancang untuk mendukung interoperabilitas dan interaksi antar sistem pada suatu jaringan. *Webservice* digunakan sebagai suatu fasilitas yang disediakan oleh suatu *website* untuk menyediakan layanan (dalam bentuk informasi) kepada sistem lain, sehingga sistem lain dapat berinteraksi dengan sistem tersebut melalui

layanan – layanan (*service*) yang disediakan oleh suatu sistem yang menyediakan *webservice*. *Webservice* menyimpan data informasi dalam format XML, sehingga data ini dapat diakses oleh sistem lain walaupun berbeda platform, sistem operasi, maupun bahasa *compiler* [14].

2.8. Flask Python

Flask Python merupakan *microframework* berbasis Python untuk membuat *webservice*. Disebutkan sebagai *microframework* karena pengembang aplikasi pada umumnya hanya menggunakan satu atau dua fungsionalitas. Flask Python sudah menyediakan pustaka untuk mendukung ekstensi yang bisa digunakan untuk menambahkan fitur pada aplikasi. Ekstensi pada Flask Python digunakan untuk mengirim dan menerima objek yang digunakan untuk komunikasi antara aplikasi dan server [15]. Aplikasi *webservice* bisa dibangun secara minimalis menggunakan Flask Python karena menggunakan fungsionalitas yang hanya dibutuhkan oleh pengembang saja. Berbasis Python karena *microframework* ini dibangun dan dikembangkan menggunakan bahasa Python. Salah satu contoh penggunaan Flask Python bisa dilihat pada Kode Sumber 2.2.

```
from flask import Flask
app = Flask(__name__)

@app.route("/")
def hello():
    return "Hello World!"

if __name__ == "__main__":
    app.run()
```

Kode Sumber 2.2. Contoh Penggunaan Flask Python

BAB III

ANALISIS DAN PERANCANGAN SISTEM

Bab ini membahas tahap analisis permasalahan dan perancangan dari sistem yang akan dibangun. Analisis topik permasalahan yang diangkat dalam pengerjaan tugas akhir. Analisis kebutuhan mencantumkan kebutuhan-kebutuhan yang diperlukan perangkat lunak. Selanjutnya dibahas mengenai perancangan sistem yang dibuat.

3.1. Analisis Perangkat Lunak

Pada subbab ini akan dibahas mengenai analisa kebutuhan perangkat lunak yang dibangun. Hal yang akan dibahas meliputi deskripsi umum perangkat lunak, spesifikasi kebutuhan perangkat lunak, dan identifikasi pengguna.

3.1.1. Deskripsi Umum Perangkat Lunak

Aplikasi media informasi nutrisi pada makanan atau produk makanan menggunakan teknologi *augmented reality* berbasis Android dibangun dengan tujuan agar kita bisa lebih memerhatikan kandungan nutrisi makanan yang akan dikonsumsi terutama kalori, karbohidrat, lemak dan protein sesuai dengan kebutuhan setiap hari. Dengan teknologi *augmented reality* diharapkan bisa lebih menarik pengguna aplikasi serta bisa menjadi media informasi nutrisi yang informatif.

Aplikasi media informasi nutrisi pada makanan atau produk makanan pada Tugas Akhir ini, menggunakan teknologi *augmented reality* berupa informasi nutrisi. Informasi yang ditampilkan berupa kandungan nutrisi yaitu kalori, lemak, karbohidrat dan protein. Misalnya pengguna aplikasi akan mengonsumsi 1 buah pisang, 1 buah jeruk dan 5 buah anggur. Untuk mengetahui kandungan nutrisi di dalam makanan atau produk makanan tersebut, pengguna bisa mendapatkan informasi

dengan mengambil foto makanan atau produk makanan. Aplikasi akan menampilkan informasi nutrisi yang disimpan di basis data dan dikemas dalam bentuk *augmented reality*. Menggunakan metode klasifikasi gambar sehingga bisa mengidentifikasi makanan atau produk makanan dari foto untuk mendapatkan informasi nama objek. Informasi nama objek yang didapat akan dikirim ke *Fatsecret Search*, untuk mendapatkan kembali daftar informasi makanan beserta rincian informasi nutrisi.

3.1.2. Spesifikasi Kebutuhan Perangkat Lunak

Kebutuhan sistem yang dibangun pada tugas akhir ini melibatkan dua hal, yakni kebutuhan fungsional maupun kebutuhan non-fungsional, dimana masing-masing berhubungan dengan keberhasilan dalam pembuatan aplikasi tugas akhir ini.

3.1.2.1. Kebutuhan Fungsional Perangkat Lunak

Pada sistem ini, terdapat beberapa kebutuhan fungsional yang mendukung untuk jalannya aplikasi. Fungsi yang terdapat dalam aplikasi ini adalah sebagai berikut.

a) Pengenalan Objek Gambar Makanan

Aplikasi mampu melakukan pengenalan objek dengan mengambil gambar foto dari makanan atau produk makanan yang kemudian diklasifikasikan untuk mendapatkan pada kelas mana gambar tersebut memiliki kemiripan. Hal yang ditekankan pada fungsionalitas ini adalah bagaimana mengetahui nama objek makanan atau produk makanan yang difoto dengan kamera *android*.

b) Identifikasi Nutrisi

Setelah nama objek gambar makanan atau produk makanan ditemukan, kemudian aplikasi mengidentifikasi kandungan nutrisi berupa kalori, karbohidrat, protein dan kalori berdasarkan nama objek. Identifikasi nutrisi dengan mencari pada basis data di *webservice*.

c) Informasi Nutrisi dalam Augmented Reality Teks

Aplikasi ini menampilkan informasi nutrisi berupa kalori, karbohidrat, protein dan lemak terkandung di dalam makanan atau produk makanan kepada pengguna perangkat bergerak berbasis Android menggunakan teknologi *augmented reality* teks sehingga lebih menarik.

3.1.2.2. Kebutuhan Non-Fungsional

Pada sistem ini, terdapat beberapa kebutuhan non-fungsional yang mendukung dan menambah performa untuk jalannya aplikasi. Fungsi yang terdapat dalam aplikasi ini adalah sebagai berikut.

a) Terhubung dengan Jaringan Internet

Jaringan internet sangat dibutuhkan dalam penggunaan aplikasi ini yaitu untuk berkomunikasi dengan *webservice*. Apabila aplikasi tidak terhubung dengan jaringan internet yang sama, pengguna tidak bisa mendapatkan informasi nama makanan yang diidentifikasi beserta nutrisi yang terkandung didalamnya.

b) Didukung dengan CPU (*Central Processing Unit*)

CPU merupakan device terpenting dalam sebuah komputer karena merupakan otak dari komputer. CPU memiliki tugas utama yaitu untuk mengolah data berdasarkan instruksi dan data dari perangkat lunak. CPU digunakan untuk mendukung kecepatan komputasi pengenalan objek makanan.

3.1.3. Identifikasi Pengguna

Pengguna yang akan terlibat hanya terdapat satu orang saja, yakni orang yang menggunakan aplikasi. Pengguna akan berkomunikasi dengan *webservice* menggunakan komunikasi *client-server*.

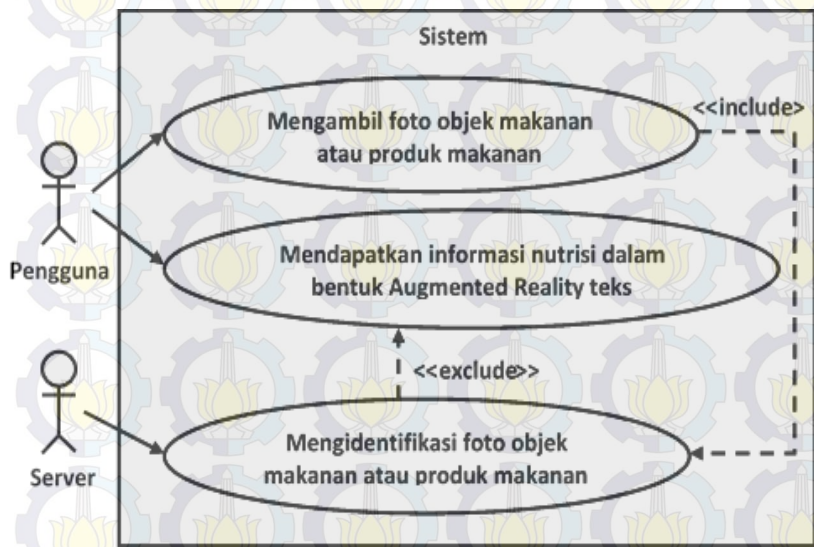
3.2. Perancangan Perangkat Lunak

Pada subbab ini membahas bagaimana rancangan dari aplikasi tugas akhir. Hal yang akan dibahas meliputi model kasus

penggunaan, definisi aktor, definisi kasus penggunaan, arsitektur umum sistem, rancangan antarmuka aplikasi, dan rancangan proses aplikasi.

3.2.1. Model Kasus Penggunaan

Dari hasil analisa deskripsi umum perangkat lunak dan spesifikasi kebutuhan yang telah dijelaskan, dapat digambarkan model kasus penggunaan aplikasi pada Gambar 3.1.



Gambar 3.1. Model Kasus Penggunaan Aplikasi

3.2.2. Definisi Aktor

Aktor yang terdapat pada aplikasi tugas akhir ini dapat dilihat pada Tabel 3.1.

Tabel 3.1. Definisi Aktor

No	Nama	Deskripsi
1	Pengguna	Merupakan aktor yang menggunakan aplikasi untuk mengambil dan

No	Nama	Deskripsi
		mengunggah foto objek makanan atau produk makanan ke server.
2	<i>Server</i>	Merupakan aktor dibalik berjalannya sistem yang mengomputasi foto objek makanan sehingga diperoleh informasi berupa nama dan nutrisi yang terkandung.

3.2.3. Definisi Kasus Penggunaan

Pada Gambar 3.1 telah dijelaskan bahwa aktor yang dalam hal ini disebut pengguna mempunyai tiga kasus penggunaan, yakni mengambil foto objek makanan atau produk makanan, mendapatkan informasi nutrisi dalam bentuk *augmented reality* teks, dan mengidentifikasi foto objek makanan atau produk makanan. Detail mengenai kasus penggunaan tersebut dapat dilihat pada Tabel 3.2.

Tabel 3.2. Definisi Kasus Penggunaan

No	Kode Kasus Penggunaan	Nama Kasus Penggunaan	Keterangan
1	KP-01	Mengambil foto objek makanan atau produk makanan.	Pengguna menggunakan makanan atau produk makanan sebagai objek yang akan difoto.
2	KP-02	Mendapatkan informasi nutrisi dalam bentuk <i>augmented reality</i> teks.	Pengguna mendapatkan rangkuman informasi nutrisi berupa kalori, karbohidrat, lemak, dan protein dalam bentuk <i>augmented reality</i> teks

No	Kode Kasus Penggunaan	Nama Kasus Penggunaan	Keterangan
			pada layar perangkat bergerak.
3	KP-03	Mengidentifikasi foto objek makanan atau produk makanan.	<i>Server</i> mengidentifikasi untuk mengenali nama objek yang difoto kemudian mencari pada basis data <i>fatsecret</i> mengenai nutrisi yang terkandung

3.2.3.1. Kasus Penggunaan Mengambil Foto Objek Makanan atau Produk Makanan

Spesifikasi kasus penggunaan mengambil foto objek makanan atau produk makanan dapat dilihat pada Tabel 3.3.

Tabel 3.3. Spesifikasi Kasus Penggunaan Mengambil Foto Objek Makanan atau Produk Makanan

Nama Kasus Penggunaan	Mengambil foto objek makanan atau produk makanan.
Nomor	KP-01
Deskripsi	Pengguna menggunakan makanan atau produk makanan sebagai objek untuk diambil gambar foto
Aktor	Pengguna
Kondisi Awal	Pengguna berada pada mode pengambilan foto di aplikasi dan terhubung dengan jaringan internet
Alur Normal	<ol style="list-style-type: none"> 1. Halaman utama muncul sebagai tanda aplikasi telah dijalankan. 2. Pengguna mengarahkan kamera ke objek makanan atau produk makanan.

	3. Pengguna menekan tombol <i>Capture Image</i> untuk mengambil foto objek 4. Foto objek dikirim ke <i>webservice</i> dengan ekstensi format berkas .jpg
Alur Alternatif	-
Kondisi Akhir	Foto objek terkirim ke <i>webservice</i>

3.2.3.2. Kasus Penggunaan Mendapatkan Informasi Nutrisi dalam Bentuk *Augmented Reality* Teks

Spesifikasi kasus penggunaan mendapatkan informasi nutrisi dalam bentuk *augmented reality* teks. dapat dilihat pada Tabel 3.4.

Tabel 3.4. Spesifikasi Kasus Penggunaan Mendapatkan Informasi Nutrisi dalam Bentuk *Augmented Reality* Teks

Nama Kasus Penggunaan	Mendapatkan informasi nutrisi dalam bentuk <i>augmented reality</i> teks.
Nomor	KP-02
Deskripsi	Pengguna dapat melihat informasi nutrisi di layar kamera berupa kalori, karbohidrat, lemak dan protein.
Aktor	Pengguna
Kondisi Awal	Informasi nama objek dan nutrisi terkandung dikembalikan ke perangkat bergerak pengguna
Alur Normal	1. Perangkat bergerak menerima informasi nama objek dan nutrisi terkandung.

	2. Perangkat menampilkan nama dan nutrisi terkandung di modus kamera aplikasi.
Alur Alternatif	A2. Nama objek makanan atau produk makanan salah 1. Pengguna mengambil foto objek kembali untuk diidentifikasi
Kondisi Akhir	Nama objek dan nutrisi berupa kalori, karbohidrat, lemak dan protein muncul di layar kamera pengguna

3.2.3.3. Kasus Penggunaan Mengidentifikasi Foto Objek Makanan atau Produk Makanan

Spesifikasi kasus penggunaan mengidentifikasi foto objek makanan atau produk makanan. dapat dilihat pada Tabel 3.5.

Tabel 3.5. Spesifikasi Kasus Penggunaan Mengidentifikasi Foto Objek Makanan atau Produk Makanan.

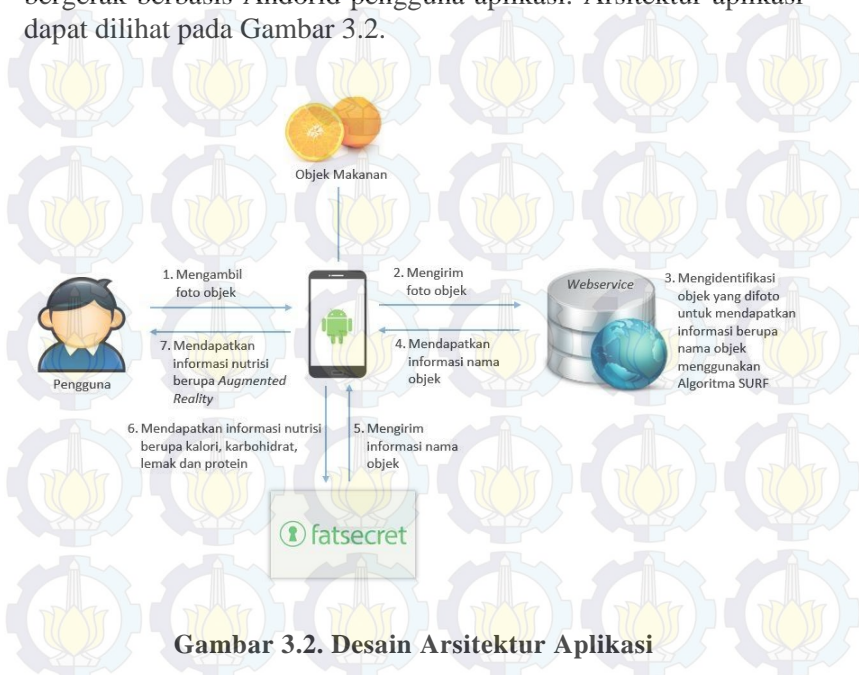
Nama Kasus Penggunaan	Mengidentifikasi foto objek makanan atau produk makanan.
Nomor	KP-03
Deskripsi	<i>Webservice</i> mengidentifikasi foto objek yang dikirim oleh aplikasi di perangkat bergerak pengguna
Aktor	<i>Webservice</i>
Kondisi Awal	<i>Webservice</i> menerima berkas dengan ekstensi .jpg dari perangkat bergerak pengguna
Alur Normal	1. <i>Webservice</i> menerima berkas dengan ekstensi .jpg 2. Ekstraksi fitur <i>keypoint</i> dari berkas .jpg

	3. Membandingkan fitur berkas .jpg dengan fitur yang disimpan di basis data. 4. Mengurutkan fitur basis data yang memiliki kemiripan pada fitur berkas .jpg berdasarkan kelas objek 5. Fitur di basis data yang memiliki kemiripan pada berkas .jpg adalah kelas objek dari berkas .jpg 6. Nama kelas objek dikirimkan ke API Platform Fatsecret 7. Fatsecret mencari kandungan nutrisi pada nama kelas objek berupa kalori, karbohidrat, protein dan lemak.
Alur Alternatif	-
Kondisi Akhir	Kandungan nutrisi berupa kalori, karbohidrat, protein dan lemak dikirimkan dalam ke perangkat bergerak pengguna

3.2.4. Arsitektur Umum Sistem

Aplikasi dibangun menggunakan arsitektur perangkat lunak yang mengintegrasikan perangkat bergerak dengan *webservice*. Dimana perangkat bergerak berbasis Android digunakan oleh pengguna aplikasi, kemudian modul kamera untuk pengambilan foto diarahkan pada objek makanan atau produk makanan. Foto objek akan dikirimkan ke *webservice*, dimana *webservice* mengidentifikasi untuk mendapatkan kembali informasi berupa nama objek. *Webservice* mengirimkan informasi berupa nama objek untuk mencari di basis data *Fatsecret* berupa informasi kalori, karbohidrat, protein dan lemak berdasarkan nama objek yang didapatkan. Informasi nutrisi berupa kalori, karbohidrat, protein dan lemak ditampilkan dalam bentuk *augmented reality*.

Rangkuman informasi nutrisi ditampilkan pada layar perangkat bergerak berbasis Andorid pengguna aplikasi. Arsitektur aplikasi dapat dilihat pada Gambar 3.2.



Gambar 3.2. Desain Arsitektur Aplikasi

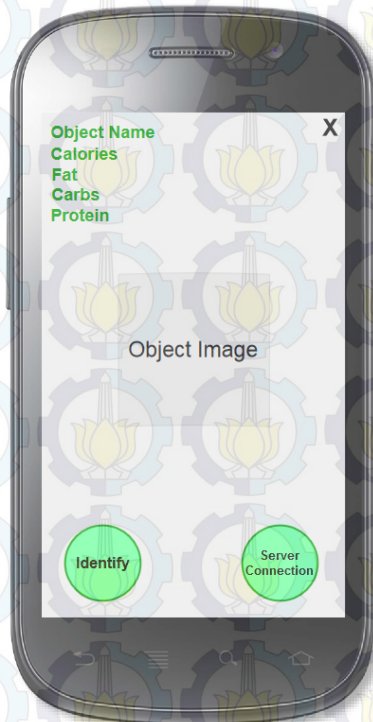
3.2.5. Rancangan Antarmuka Aplikasi

Rancangan antarmuka aplikasi diperlukan untuk memberikan gambaran umum kepada pengguna mengenai bagaimana sistem yang ada dalam aplikasi ini berinteraksi dengan pengguna.

3.2.5.1. Rancangan Antarmuka Halaman Utama

Pada halaman utama aplikasi akan menampilkan modus kamera yang siap untuk pengambilan gambar foto objek makanan atau produk makanan. Pada halaman ini terdapat tombol *Identify* untuk mengambil gambar objek makanan serta menampilkan kandungan nutrisi makanan. Tombol *Server Connection* berfungsi

untuk mengatur Alamat IP pada jaringan yang terhubung dengan aplikasi. Rancangan antarmuka halaman utama dilihat pada Gambar 3.3.



Gambar 3.3. Rancangan Antarmuka Halaman Utama

3.2.5.2. Rancangan Antarmuka Pengiriman Gambar ke Webservice

Rancangan antarmuka pengiriman gambar ke *webservice* akan menampilkan pesan peringatan bahwa gambar yang telah diambil sedang dikirim ke *server* untuk diidentifikasi. Gambar objek yang dikirimkan berupa berkas dengan ekstensi *.jpg*. Pengiriman gambar ke *server* berguna untuk mendapatkan nama

dari objek yang difoto dan rangkuman nutrisi yang terkandung pada objek. Rancangan antarmuka dapat dilihat pada Gambar 3.4.

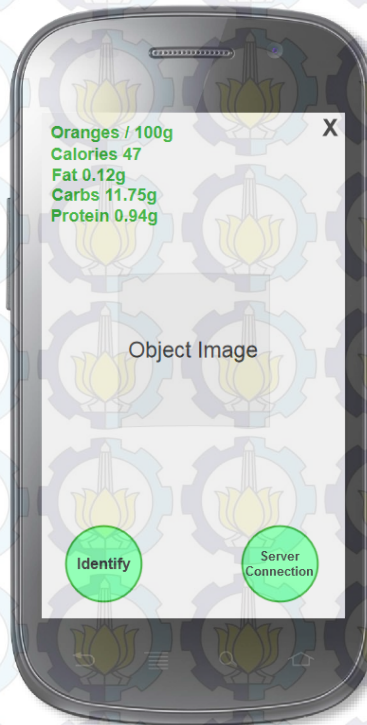


Gambar 3.4. Rancangan Antarmuka Pengiriman Gambar ke Server

3.2.5.3. Rancangan Antarmuka Rangkuman Nutrisi

Pada halaman ini aplikasi akan menampilkan rangkuman nutrisi pada layar modus kamera di antarmuka aplikasi. Tombol *Identify* berguna untuk mengambil gambar objek serta menampilkan rangkuman nutrisi yang terkandung pada objek makanan yang difoto oleh pengguna. Rangkuman nutrisi

didapatkan dari hasil pencarian di *server* basis data *Fatsecret*. Rangkuman nutrisi berupa nama objek makanan beserta jumlah takaran, kalori, lemak, karbohidrat dan protein. Rancangan antarmuka dapat dilihat pada Gambar 3.5.



Gambar 3.5. Rancangan Antarmuka Rangkuman Nutrisi

3.2.5.4. Rancangan Antarmuka *Server Connection*

Rancangan antarmuka *server connection* berguna sebagai pengaturan koneksi jaringan internet yang terhubung dengan aplikasi dan *webservice*. Aplikasi dan *webservice* harus terhubung dalam satu jaringan koneksi yang sama. Terdapat tombol *server*

address untuk mengakses halaman permukaan untuk memasukkan alamat IP *server*. Rancangan antarmuka dapat dilihat pada Gambar 3.6.



Gambar 3.6. Rancangan Antarmuka *Server Connection*

3.2.5.5. Rancangan Antarmuka Masukkan Alamat IP *Server*

Pada rancangan antar muka ini berfungsi untuk memasukkan alamat IP yang terhubung pada aplikasi. Alamat IP *server* yang dimasukkan tergantung pada koneksi yang terhubung antara aplikasi dan *webservice*. Antar muka ini memberikan manfaat kepada pengguna agar bisa dinamis mengganti alamat IP *server*. Dengan mengetikkan sendiri pada alamat IP *server*, pengguna bisa

memasukkan langsung alamat IP *server* yang terhubung antara aplikasi dan *webservice*. Rancangan antarmuka dapat dilihat pada Gambar 3.7.

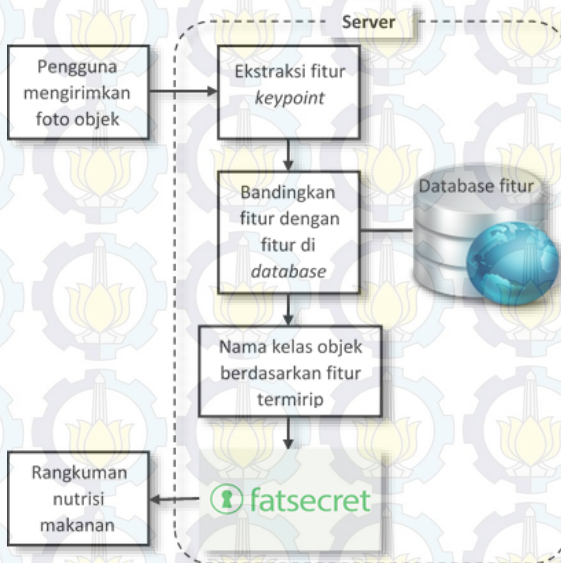


Gambar 3.7. Rancangan Antarmuka Masukkan Alamat IP Server

3.2.5.6. Rancangan Antarmuka *Webservice*

Rancangan antarmuka *webservice* berguna untuk mendapatkan rangkuman nutrisi berdasarkan foto objek yang diidentifikasi. *Webservice* menerima masukan berupa gambar. Kemudian gambar diekstraksi fitur berdasarkan fitur keypoint. Lalu fitur gambar dibandingkan dengan fitur di basis data.

Selanjutnya nama kelas objek berdasarkan fitur termirip dikirim ke *Fatsecret* untuk mengetahui kandungan nutrisi yang terkandung pada objek gambar. Fatsecret sudah terdapat basis data makanan beserta kandungan nutrisinya. Setelah itu rangkuman nutrisi berupa kalori, lemak, karbohidrat, dan protein dikirimkan ke perangkat bergerak pengguna. Rancangan antarmuka dapat dilihat pada Gambar 3.8.



Gambar 3.8. Rancangan Antarmuka Webservice

3.2.6. Rancangan Proses Penggunaan Aplikasi

Dalam proses ini terdapat proses dimana pengguna menggunakan aplikasi hingga mendapatkan informasi nutrisi yang terkandung di dalam objek makanan. Pengguna mengirimkan foto objek makanan dengan ukuran 400 x 400 piksel. Pada lapisan *server* terdapat basis data gambar untuk masing-masing kelas sejumlah 20 gambar. Gambar pada basis data, masing-masing berukuran 400 x 400 piksel.



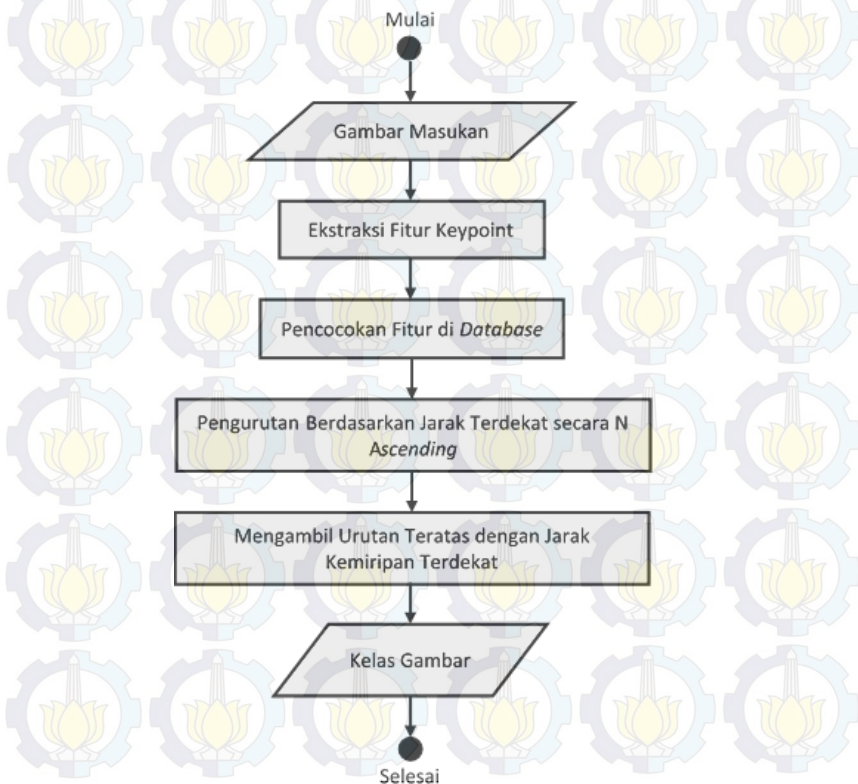
Gambar 3.9. Diagram Alir Proses Penggunaan Aplikasi

Untuk mendapatkan fitur gambar, ekstraksi fitur menggunakan pustaka SURF (*Speeded-up Robust Features*) yang sudah disediakan oleh OpenCV. Setelah didapatkan fitur gambar

berupa *keypoint*, kemudian dihitung kemiripan fitur menggunakan *Euclidian Distance* dengan Persamaan 2.1. Rancangan proses penggunaan aplikasi dapat dilihat pada Gambar 3.9.

3.2.7. Rancangan Proses Klasifikasi Gambar

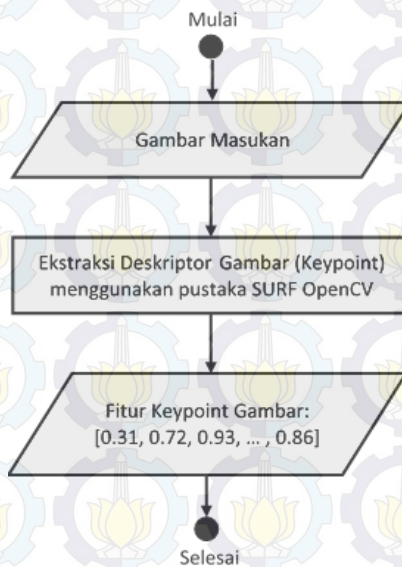
Proses ini sangat dibutuhkan dalam berjalannya aplikasi. Proses klasifikasi gambar berguna untuk menentukan gambar termirip di basis data gambar masukan yang dikirimkan oleh perangkat bergerak. Secara umum proses klasifikasi gambar dapat dilihat pada Gambar 3.10.



Gambar 3.10. Diagram Alir Proses Klasifikasi Gambar

3.2.7.1. Rancangan Proses Ekstraksi Fitur *Keypoint* Gambar

Proses ini sangat dibutuhkan dalam berjalannya aplikasi, karena pada proses ekstraksi fitur *keypoint* terdapat pengambilan deskriptor yang diperlukan untuk menentukan kelas objek gambar atau citra di basis data. Proses ekstraksi fitur menggunakan pustaka SURF (*Speeded-Up Robust Features*) yang sudah disediakan oleh OpenCV. Citra gambar masukan yang dikirim dari Android ke *Webservice* akan melalui proses pengambilan fitur deskriptor. Fitur deskriptor yang diambil berupa *keypoint* yang kemudian disimpan dalam berkas berekstensi .pkl. Rancangan proses ekstraksi fitur *keypoint* gambar bisa dilihat pada Gambar 3.11.

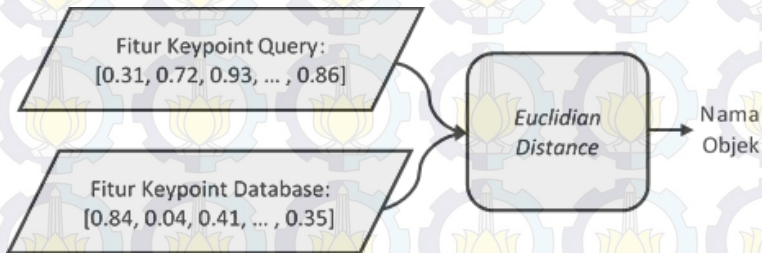


Gambar 3.11. Proses Ekstraksi Fitur *Keypoint* Gambar

3.2.7.2. Rancangan Proses Perhitungan Kemiripan Gambar

Proses perhitungan kemiripan gambar sangat dibutuhkan dalam menentukan kelas objek gambar. Dilakukan pendeskripsian

titik-titik fitur menjadi deskriptor vektor agar titik-titik fitur memiliki ketahanan terhadap rotasi, kontras, dan perubahan sudut pandang. Setiap fitur yang terdeteksi akan diberikan orientasi agar tahan terhadap rotasi. Fitur vektor deskriptor yang digunakan pada SURF (*Speeded-Up Robust Features*) ini berupa *keypoint* sejumlah 400 dengan *hessian-threshold* sejumlah 100. *Hessian-threshold* adalah ambang batas fitur pengambilan *keypoint* untuk perhitungan kemiripan. *Keypoint* sebagai fitur deskriptor dapat dihitung kemiripannya dengan menggunakan fungsi *Euclidian Distance* tanpa dilakukan normalisasi fitur. Perhitungan menggunakan *Euclidian Distance* dapat dilihat pada Persamaan 2.1. Semakin kecil jarak *Euclidian Distance* maka semakin besar kemiripan fitur gambar. Keluaran pada proses perhitungan kemiripan gambar ini berupa nama objek makanan atau produk makanan. Proses perhitungan kemiripan gambar bisa dilihat pada Gambar 3.12.



Gambar 3.12. Proses Perhitungan Kemiripan Gambar

BAB IV

IMPLEMENTASI

Bab ini membahas tentang implementasi dari perancangan sistem. Bab ini berisi proses implementasi dari setiap perancangan pada bab sebelumnya. Namun, pada hasil akhir mungkin saja terjadi perubahan kecil. Implementasi pembangunan *webservice* menggunakan bahasa Python 2.7 dan untuk *client* Android menggunakan Android Studio.

4.1. Lingkungan Pembangunan

Dalam pembangunan aplikasi, digunakan beberapa perangkat sebagai pendukung baik perangkat keras maupun perangkat lunak. Lingkungan pembangunan dijelaskan sebagai berikut.

4.1.1. Lingkungan Pembangunan Perangkat Keras

Perangkat keras yang digunakan dalam pembuatan aplikasi ini adalah sebuah laptop Toshiba Satellite L745 14-inch dan perangkat bergerak sebagai alat uji coba dengan spesifikasi sebagai berikut.

- Prosesor Intel(R) Core(TM) i5 - 2410M CPU @ 2,30GHz
- (RAM) 4,00 GB
- NVIDIA GeForce GT 525M
- Asus Zenfone 4 Android OS 4.3 (Jelly Bean) RAM 1GB Camera 5MP

4.1.2. Lingkungan Pembangunan Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam pembuatan aplikasi adalah sebagai berikut.

- Python 2.7
- Sublime Text 3
- Android Studio 1.2
- Windows 7 Professional 64-bit

4.2. Implementasi Antarmuka Aplikasi Android

Pada subbab ini akan dibahas mengenai hasil implementasi yang dilakukan berdasarkan rancangan antarmuka. Nantinya akan digunakan kamera perangkat bergerak Android untuk pengambilan gambar objek makanan atau produk makanan yang kemudian diproses oleh *webservice* untuk mendapatkan kandungan nutrisi. Implementasi Antarmuka pada Android menggunakan Android Studio, sedangkan *webservice* menggunakan Sublime Text 3 menggunakan bahasa pemrograman Python 2.7.

4.2.1. Implementasi Antarmuka Halaman Utama

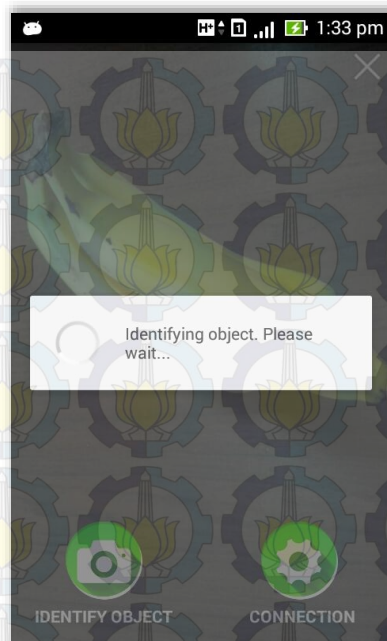
Antarmuka Halaman Utama adalah antarmuka yang pertama kali akan muncul pada saat pengguna membuka aplikasi. Antarmuka ini berguna untuk mengambil gambar dari objek makanan atau produk makanan. Pertama kali aplikasi dijalankan maka akan berada pada modus pengambilan gambar objek. Terdapat tombol *identify object* untuk mengambil gambar objek makanan serta menampilkan kandungan nutrisi makanan. Tombol *connection* berfungsi untuk mengatur alamat IP pada jaringan yang terhubung dengan aplikasi. Pada layar bagian kanan atas terdapat tombol untuk keluar dari aplikasi. Implementasi dari antarmuka dapat dilihat pada Gambar 4.1.



Gambar 4.1. Antarmuka Halaman Utama

4.2.2. Implementasi Antarmuka Pengiriman Gambar ke Webservice

Antarmuka pengiriman gambar ke *webservice* adalah sebuah antarmuka yang disediakan aplikasi untuk menampilkan pesan peringatan bahwa gambar yang telah diambil sedang dikirim ke *server* untuk diidentifikasi. Gambar objek makanan atau produk makanan yang dikirimkan berupa berkas dengan ekstensi *.jpg*. Setelah *server* menerima gambar yang dikirimkan oleh Android, gambar diidentifikasi untuk mendapatkan nama dan nutrisi yang terkandung pada makanan atau produk makanan. Implementasi dari antarmuka dapat dilihat pada Gambar 4.2.



Gambar 4.2. Antarmuka Pengiriman Gambar ke Webservice

4.2.3. Implementasi Antarmuka Rangkuman Nutrisi

Antarmuka rangkuman nutrisi akan menampilkan informasi nutrisi pada layar modus kamera di antarmuka aplikasi. Apabila ingin mengidentifikasi objek kembali, pengguna bisa menggunakan tombol *identify object* pada layar kamera. Rangkuman nutrisi didapatkan dari hasil pencarian di *server* basis data *Fatsecret*. Rangkuman nutrisi berupa nama objek makanan beserta jumlah takaran untuk setiap konsumsi, kalori, lemak, karbohidrat dan protein.. Implementasi dari antarmuka dapat dilihat pada Gambar 4.3.



Gambar 4.3. Antarmuka Rangkuman Nutrisi

4.2.4. Implementasi Antarmuka *Server Connection*

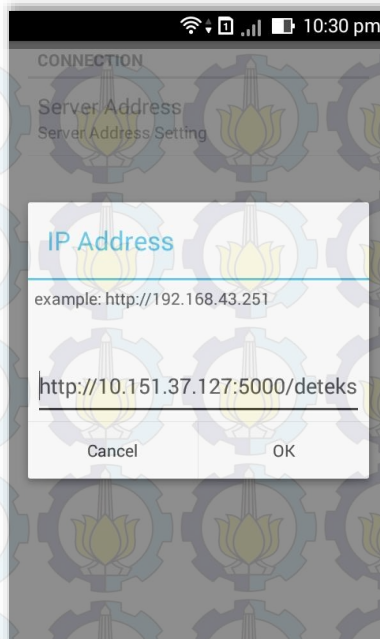
Antarmuka *server connection* menampilkan tampilan berupa layar aktifitas untuk pengaturan alamat IP *server*. Terdapat teks *connection* sebagai nama dari aktifitas layar. Dan tombol *server address* untuk menampilkan halaman permukaan memasukkan alamat IP *server*. Implementasi antarmuka dapat dilihat pada Gambar 4.4.



Gambar 4.4. Antarmuka *Server Connection*

4.2.5. Implementasi Antarmuka Masukkan Alamat IP Server

Antarmuka masukkan alamat IP server berfungsi untuk menghubungkan antara aplikasi dengan *webservice*. Aplikasi harus terhubung dengan jaringan internet yang sama sehingga konfigurasi alamat IP server yang terhubung antar keduanya menggunakan jaringan lokal yang sama. Implementasi antar muka dapat dilihat pada Gambar 4.5.



Gambar 4.5. Antarmuka Masukkan Alamat IP Server

4.3. Implementasi Antarmuka *Webservice*

Pada subbab ini akan dibahas mengenai implementasi aplikasi pada bagian *webservice*. Dijelaskan juga mengenai fungsi – fungsi yang dibutuhkan untuk menunjang aplikasi agar dapat berjalan sebagaimana mestinya. Implementasi ini dilakukan menggunakan Sublime Text 3 dengan bahasa pemrograman Python 2.7.

4.3.1. Implementasi Lapisan *Server* pada *Webservice*

Implementasi *webservice* menggunakan pustaka Flask Python dengan metode POST dan GET. Metode POST dan GET digunakan untuk komunikasi pengiriman data berupa gambar

objek makanan antara Android dan *webservice*. Ketika Android mengirimkan file berupa gambar objek makanan, *webservice* akan menyimpan file pada direktori *testing* yang kemudian diekstraksi fitur dalam berkas .pkl. Ekstraksi fitur yang disimpan akan diproses untuk dihitung kemiripan dengan berkas .pkl pada direktori basis data. Untuk lebih detailnya dapat dilihat pada Kode Sumber 4.1.

```
app = Flask(__name__)
app.debug = True

@app.route("/deteksi", methods=['GET', 'POST'])
def deteksi():
    if request.method == 'POST':
        file = request.files['file']
        file.save('testing/' +
file.filename)
        (kelas,info) =
sim.deteksi_image(file.filename)
        return str(kelas)+"#"+str(info)

if __name__ == "__main__":
    app.run(host = '0.0.0.0', debug = True)
```

Kode Sumber 4.1. Implementasi Server pada Webservice

Koneksi Android dan *webservice* menggunakan *localhost connection* sehingga aplikasi yang digunakan harus terhubung jaringan lokal untuk komunikasi.

4.3.1.1. Ekstraksi Fitur Keypoint

Pengambilan fitur pada gambar tidak menggunakan ekstraksi fitur primitif seperti warna, bentuk, luasan dan lain sebagainya. Ekstraksi fitur yang diambil berupa *keypoint* yaitu *interest point* pada kandidat piksel yang menyimpan piksel terunik sebagai ciri khas objek gambar. *Keypoint* kemudian dijadikan sebagai deskriptor untuk proses pencocokan gambar. Proses ekstraksi fitur ini menggunakan pustaka OpenCV Python. Untuk lebih detailnya dapat dilihat pada Kode Sumber 4.2.

```

def save_feature(keypoints, descriptors,
filename):
    i = 0
    temp_array = []
    for point in keypoints:
        temp = (point.pt, point.size,
point.angle, point.response, point.octave,
point.class_id, descriptors[i])
        i += 1
        temp_array.append(temp)
    s = pickle.dumps(temp_array)
    open(filename, 'w').write(s)

```

Kode Sumber 4.2. Implementasi Ekstraksi Fitur *Keypoint*

4.3.1.2. Menyimpan Hasil Ekstraksi Fitur

Pada fungsi menyimpan hasil ekstraksi fitur dilakukan pengambilan *keypoint* sebagai deskriptor untuk semua gambar di basis data direktori. Ambang batas deskriptor gambar menggunakan *hessian_threshold* dengan ketentuan apabila ambang semakin kecil maka deskriptor yang diambil akan banyak. Sebaliknya, apabila ambang semakin besar maka deskriptor yang diambil semakin sedikit. Pengambilan ekstraksi fitur menggunakan SURF OpenCV. Hasil ekstraksi fitur kemudian disimpan dalam berkas .pkl. Untuk lebih detailnya dapat dilihat pada Kode Sumber 4.3.

```

def comp_feature():
    images_train = [f for f in
os.listdir('.') if '.jpg' in f]
    hessian_threshold = 100
    detector = cv2.SURF(hessian_threshold)
    for image in images_train:
        img = cv2.imread(image)
        img_gray = cv2.cvtColor(img,
cv2.COLOR_BGR2GRAY)

```

```

        keypoints, desc =
detector.detectAndCompute(img_gray, None)
        save_feature(keypoints, desc,
image[:-4] + '.pkl')

```

Kode Sumber 4.3. Implementasi Penyimpanan Hasil Ekstraksi Fitur

4.3.1.3. Membaca Ekstraksi Fitur Gambar dari Android Untuk Dikomputasi

Pada fungsi ini terdapat proses pembacaan hasil ekstraksi gambar yang dikirim ke *webservice*. Hasil ekstraksi berupa deskriptor yang digunakan sebagai pencocokan gambar pada basis data di direktori. Pembacaan ekstraksi fitur mengambil vektor piksel gambar. Untuk gambar di basis data dikelompokkan berdasarkan nama objek sehingga hasil ekstraksi fitur berupa deskriptor yang tersimpan berdasarkan nama gambar. Deskriptor yang diambil disimpan ke berkas sementara yang kemudian dilakukan pencocokan di basis data direktori. Untuk lebih detail dapat dilihat pada Kode Sumber 4.4.

```

def load_feature(filename):
    features = pickle.loads(open(filename,
'r').read())
    keypoints, descriptors = [], []
    for feature in features:
        temp_feature =
cv2.KeyPoint(x=feature[0][0], y=feature[0][1],
_size=feature[1], _angle=feature[2],
_response=feature[3], _octave=feature[4],
_class_id=feature[5])
        temp_descriptor = feature[6]
        keypoints.append(temp_feature)
        descriptors.append(temp_descriptor)
    return keypoints, np.array(descriptors)

def comp_feature():
    images_train = [f for f in
os.listdir('.') if '.jpg' in f]

```



```

hessian_threshold = 100
detector = cv2.SURF(hessian_threshold)
for image in images_train:
    img = cv2.imread(image)
    img_gray = cv2.cvtColor(img,
cv2.COLOR_BGR2GRAY)
    keypoints, desc =
detector.detectAndCompute(img_gray, None)
    save_feature(keypoints, desc,
image[:-4] + '.pkl')

```

Kode Sumber 4.4. Implementasi Pembacaan Hasil Ekstraksi Fitur Gambar dari Android Untuk Dikomputasi

4.3.1.4. Mencocokkan Gambar di Basis Data Direktori

Pada fungsi ini terdapat proses pencocokan gambar yang dikirim dari Android ke *webservice*. Gambar yang dikirim dari Android sebelumnya diambil ekstraksi fitur terlebih dahulu kemudian dikomputasi dengan fitur – fitur setiap kelas objek di basis data direktori. Untuk setiap fitur gambar di basis data direktori dihitung nilai kemiripannya dengan gambar yang dikirim oleh Android. Semakin tinggi nilai kemiripan maka akan semakin tinggi juga untuk menjadi kandidat gambar termirip. Untuk lebih detail dapat dilihat pada Kode Sumber 4.5.

```

def deteksi_image(filename):
    images_train = [f for f in
os.listdir('.') if '.pkl' in f]
    hessian_threshold = 100
    detector = cv2.SURF(hessian_threshold)
    img = cv2.imread('testing/' + filename)
    img_gray = cv2.cvtColor(img,
cv2.COLOR_BGR2GRAY)
    keypoints, desc =
detector.detectAndCompute(img_gray, None)
    save_feature(keypoints, desc, 'testing/'
+ filename[:-4] + '.pkl')
    similarity = 0

```

```

for image2 in images_train:
    val = compute_score2(
'testing/' + filename[:-4] + '.pkl', image2)
    if similarity < val:
        similarity = val
        class_name = image2
    print
image2,similarity
print(class_name[:-6])

```

Kode Sumber 4.5. Implementasi Pencocokan Gambar di Basis data Direktori

4.3.1.5. Mendapatkan Rangkuman Nutrisi Objek Makanan

Pada fungsi ini terdapat *consumer key* dan *consumer secret* untuk menggunakan REST API pada *Fatsecret*. Untuk mendapatkan rangkuman nutrisi berupa kalori, lemak, karbohidrat dan protein menggunakan *food search*. Masukan dari *Fatsecret* ini berupa hasil klasifikasi gambar. Klasifikasi gambar dilakukan untuk mendapatkan nama objek gambar. Nama objek gambar tersebut yang kemudian digunakan sebagai kata kunci pencarian nutrisi di *Fatsecret*. Keluaran dari *Fatsecret* berupa kandungan nutrisi kalori, lemak, karbohidrat dan protein untuk setiap takaran penyajian. Untuk lebih detail dapat dilihat pada Kode Sumber 4.6.

```

consumer_key='e0a7c995c2cf4e49973d316e4f
032a67'
consumer_secret='8ac02431-eccb-40f5-
bf89-66ee7c765e51'
fs=Fatsecret(consumer_key,consumer_secre
t)
foods = fs.foods_search(class_name[:-6])
print foods[0]['food_description']
return(class_name[:-
6],foods[0]['food_description'])

```

Kode Sumber 4.6. Implementasi Mendapatkan Rangkuman Nutrisi Objek Makanan

BAB V

PENGUJIAN DAN EVALUASI

Bab ini membahas pengujian dan evaluasi pada aplikasi yang dikembangkan. Pengujian yang dilakukan adalah pengujian terhadap kebutuhan fungsional secara keseluruhan. Pengujian ini mengacu pada kasus penggunaan pada bab tiga. Pengujian dilakukan dengan beberapa skenario. Hasil evaluasi menjabarkan tentang rangkuman hasil pengujian pada bagian akhir bab ini.

5.1. Lingkungan Pembangunan

Dalam membangun aplikasi ini digunakan beberapa perangkat pendukung baik perangkat keras maupun perangkat lunak. Perangkat keras yang digunakan dalam pembuatan aplikasi ini adalah sebuah laptop Toshiba Satellite L745 14-inch dan perangkat bergerak sebagai alat uji coba dengan spesifikasi sebagai berikut.

- Prosesor Intel(R) Core(TM) i5 - 2410M CPU @ 2,30GHz
- (RAM) 4,00 GB
- NVIDIA GeForce GT 525M
- Asus Zenfone 4 Android OS 4.3 (Jelly Bean) RAM 1GB Camera 5MP

5.2. Skenario Pengujian

Skenario pengujian yang dilakukan adalah sebagai berikut.

1. Pengujian skenario 1 berupa pengujian terhadap kemampuan modus kamera dalam posisi siap untuk pengambilan gambar objek makanan.
2. Pengujian skenario 2 berupa pengujian terhadap penggunaan fungsionalitas pada tombol di layar modus kamera.

3. Pengujian skenario 3 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada buah pisang.
4. Pengujian skenario 4 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada buah jeruk.
5. Pengujian skenario 5 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada buah apel.
6. Pengujian skenario 6 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Indomie.
7. Pengujian skenario 7 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Pringles.
8. Pengujian skenario 8 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada buah Lemon .
9. Pengujian skenario 9 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Pocky.
10. Pengujian skenario 10 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Oreo.
11. Pengujian skenario 11 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Snickers.
12. Pengujian skenario 12 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Tango.
13. Pengujian skenario 13 berupa kalkulasi kebenaran aplikasi dalam mengidentifikasi nama dan nutrisi pada objek makanan atau produk makanan.
14. Pengujian skenario 14 berupa pengujian terhadap pengguna secara langsung yaitu dengan melibatkan

Mahasiswa Teknik Informatika dan kalangan masyarakat sekitar ITS sebagai pengguna langsung aplikasi.

5.2.1. Pengujian Skenario 1 dan Evaluasi

Pada pengujian skenario 1 ini akan dilakukan pengujian terhadap modus pengambilan gambar objek makanan atau produk makanan. Modus pengambilan gambar merupakan halaman utama yang akan keluar ketika aplikasi dijalankan. Pada modus kamera Terdapat dua tombol utama dalam menjalankan fungsionalitas aplikasi yaitu tombol *identify object* dan tombol *connection*. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.1.

Tabel 5.1. Skenario Pengujian 1

Nama Skenario Pengujian	Pengujian Modus Pengambilan Gambar
Kode	SP-01
Tujuan Pengujian	Menguji modus pengambilan gambar makanan atau produk makanan untuk diidentifikasi.
Kondisi Awal	Aplikasi terhubung pada jaringan internet yang sama dengan <i>webservice</i>
Data Input	-
Prosedur Pengujian	1. Siapkan perangkat bergerak yang sudah terpasang aplikasi. 2. Jalankan <i>webservice</i> pada perangkat keras di laptop.
Hasil yang Diharapkan	Setelah aplikasi dijalankan, halaman utama yaitu modus kamera dalam kondisi siap untuk mengambil gambar.

Hasil yang Diperoleh	Halaman utama yaitu modus kamera dalam kondisi siap untuk mengambil gambar.
----------------------	---

Pada awal pengujian, jalankan aplikasi pada perangkat bergerak. Letakan objek makanan atau produk makanan yang akan diidentifikasi. Arahkan modus kamera pada objek makanan atau produk makanan. Modus kamera pada aplikasi siap untuk proses pengambilan gambar. Terdapat tombol *identify object* untuk identifikasi gambar yang diambil pada objek dan tombol *connection* untuk pengaturan koneksi alamat IP jaringan internet yang terhubung. Proses pengujian terhadap modus pengambilan gambar objek makanan atau produk makanan dapat dilihat pada Gambar 5.1.



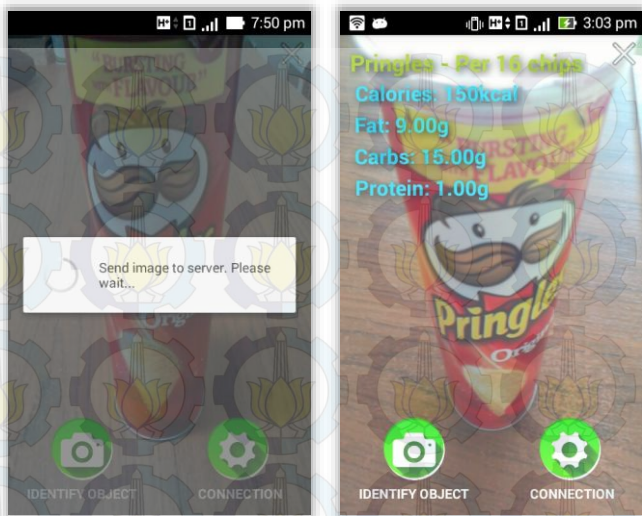
Gambar 5.1. Modus Pengambilan Gambar Objek Makanan atau Produk Makanan

5.2.2. Pengujian Skenario 2 dan Evaluasi

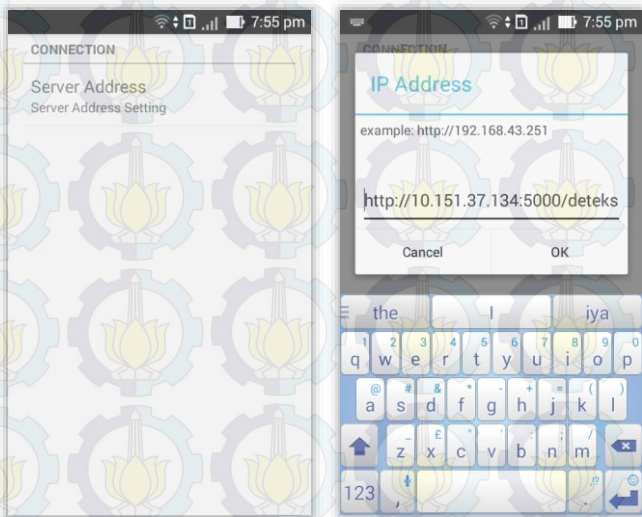
Pada pengujian skenario 2 ini akan dilakukan pengujian terhadap penggunaan fungsionalitas pada tombol di layar modus kamera. Pengujian terhadap penggunaan fungsionalitas yang dimaksud berkaitan dengan kesesuaian peran fitur tombol - tombol pada aplikasi. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.2.

Tabel 5.2. Skenario Pengujian 2

Nama Skenario Pengujian	Pengujian Fungsionalitas pada Tombol di Layar Modus Kamera
Kode	SP-02
Tujuan Pengujian	Tombol <i>identify object</i> dan <i>connection</i> berjalan sesuai dengan perannya.
Kondisi Awal	Pengguna sudah menjalankan aplikasi pada perangkat bergerak.
Data Input	-
Prosedur Pengujian	1. Pengguna berada pada modus pengambilan gambar. 2. Pengguna siap untuk mencoba fitur pada aplikasi.
Hasil yang Diharapkan	Tombol <i>identify object</i> dan <i>connection</i> berjalan sesuai dengan peran fungsinya.
Hasil yang Diperoleh	Tombol <i>identify object</i> dan <i>connection</i> berhasil berjalan sesuai dengan peran fungsinya.



Gambar 5.2. Pengujian Tombol *Identify Object*



Gambar 5.3. Pengujian Tombol *Connection*

Sesuai dengan pengujian yang telah dilakukan seperti pada Gambar 5.2. dan Gambar 5.3., dapat disimpulkan bahwa penggunaan tombol pada aplikasi sudah berjalan sesuai dengan peran fungsinya. Ketika tombol *identify object* dipilih maka aplikasi akan mengambil gambar objek untuk dikirimkan ke *webservice* agar diperoleh nama dan kandungan nutrisi pada objek makanan. Tombol *connection* berjalan sesuai dengan peran fungsinya yaitu melakukan pengaturan alamat IP jaringan internet yang terhubung antara aplikasi dan *webservice*.

5.2.3. Pengujian Skenario 3 dan Evaluasi

Pada pengujian skenario 3 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada buah pisang. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.3.

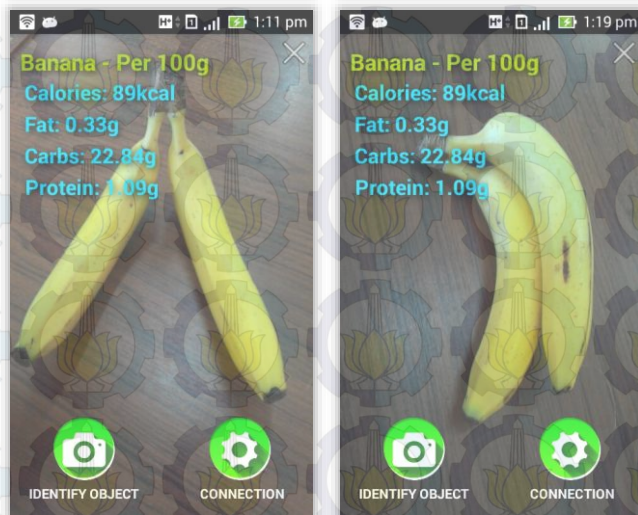
Tabel 5.3. Skenario Pengujian 3

Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Buah Pisang
Kode	SP-03
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada buah pisang.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaanodus pengambilan gambar.
Data Input	Objek buah pisang.

Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi buah pisang sebanyak 20 kali. 2. Terdapat 4 macam buah pisang dengan masing–masing buah dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi buah pisang didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi buah pisang memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi buah pisang benar sebanyak 20 dari 20 kali uji coba

Tabel 5.4. Dokumentasi Hasil Pengujian 3

Uji Coba	Hasil Identifikasi	Waktu (detik)	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	Pisang	10	11	Pisang	10
2	Pisang	11	12	Pisang	9
3	Pisang	12	13	Pisang	11
4	Pisang	11	14	Pisang	5
5	Pisang	9	15	Pisang	7
6	Pisang	10	16	Pisang	9
7	Pisang	8	17	Pisang	8
8	Pisang	11	18	Pisang	10
9	Pisang	10	19	Pisang	11
10	Pisang	9	20	Pisang	7
Tingkat kebenaran dari 20 uji coba = 100%					
Rata-rata waktu identifikasi = 9.4 detik					



Gambar 5.4. Contoh Uji Coba Pengujian Aplikasi pada Buah Pisang

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi buah pisang benar sebanyak 20 dari 20 kali uji coba. Sehingga tingkat kebenaran dari 20 kali uji coba adalah 100%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 9.4 detik. Pengujian dilakukan terhadap 4 macam buah pisang dengan modus pengambilan gambar berbeda. Contoh uji coba pengujian aplikasi pada buah pisang dapat dilihat pada Gambar 5.4.

5.2.4. Pengujian Skenario 4 dan Evaluasi

Pada pengujian skenario 4 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada buah jeruk. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama

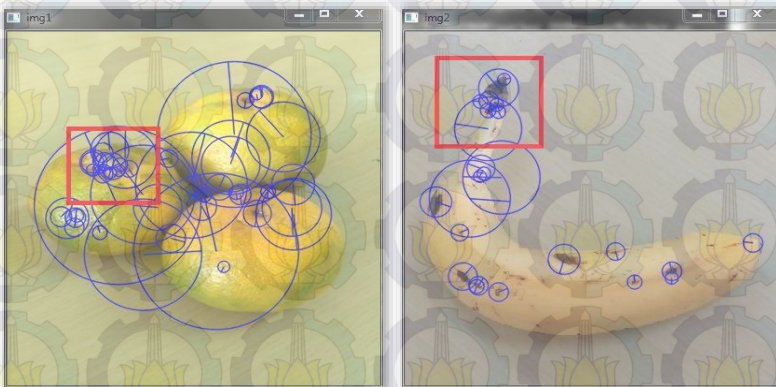
dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.5.

Tabel 5.5. Skenario Pengujian 4

Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Buah Jeruk
Kode	SP-04
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada buah jeruk.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaan modus pengambilan gambar.
Data Input	Objek buah jeruk
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi buah jeruk sebanyak 20 kali. 2. Terdapat 4 macam buah jeruk dengan masing – masing buah dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi buah jeruk didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi buah jeruk memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi buah jeruk benar sebanyak 18 dari 20 kali uji coba

Tabel 5.6. Dokumentasi Hasil Pengujian 4

Uji Coba	Hasil Identifikasi	Waktu (detik)	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	Jeruk	7	11	Jeruk	7
2	Jeruk	9	12	Jeruk	8
3	Jeruk	8	13	Jeruk	7
4	Jeruk	10	14	Jeruk	10
5	Jeruk	10	15	Jeruk	9
6	Jeruk	9	16	Pisang	11
7	Jeruk	7	17	Jeruk	8
8	Jeruk	8	18	Jeruk	9
9	Jeruk	9	19	Jeruk	10
10	Pisang	10	20	Jeruk	7
Tingkat kebenaran dari 20 uji coba = 90%					
Rata-rata waktu identifikasi = 8.65 detik					

**Gambar 5.5. Identifikasi Kemiripan Buah Jeruk dan Pisang**

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi buah jeruk benar sebanyak 18 dari 20 kali uji coba. Sedangkan 2 uji coba teridentifikasi sebagai buah pisang, hal ini dikarenakan *keypoint* pada buah jeruk memiliki

kemiripan dengan buah pisang ketika posisi pengambilan gambar untuk diidentifikasi pada uji coba 10 dan 16. Identifikasi kemiripan buah jeruk dan pisang dapat dilihat pada Gambar 5.5.



Gambar 5.6. Contoh Uji Coba Pengujian Aplikasi pada Buah Jeruk

Fitur yang diambil hanya *keypoint* saja serta bukan fitur warna dan bentuk, maka buah pisang masih terdeteksi kemiripan dengan buah jeruk. Pengujian dilakukan terhadap 4 macam buah jeruk dengan modus pengambilan gambar berbeda. Sehingga tingkat kebenaran dari 20 kali uji coba adalah 90%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 8.65 detik. Contoh uji coba pengujian aplikasi pada buah jeruk dapat dilihat pada Gambar 5.6.

5.2.5. Pengujian Skenario 5 dan Evaluasi

Pada pengujian skenario 5 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada buah

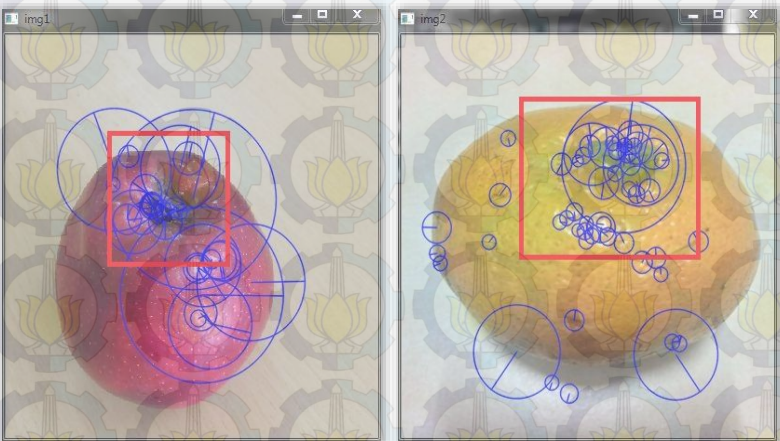
apel. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.7.

Tabel 5.7. Skenario Pengujian 5

Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Buah Apel
Kode	SP-05
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada buah apel.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaan modus pengambilan gambar.
Data Input	Objek buah apel.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi buah apel sebanyak 20 kali. 2. Terdapat 4 macam buah apel dengan masing – masing buah dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi buah apel didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi buah apel memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi buah apel benar sebanyak 14 dari 20 kali uji coba

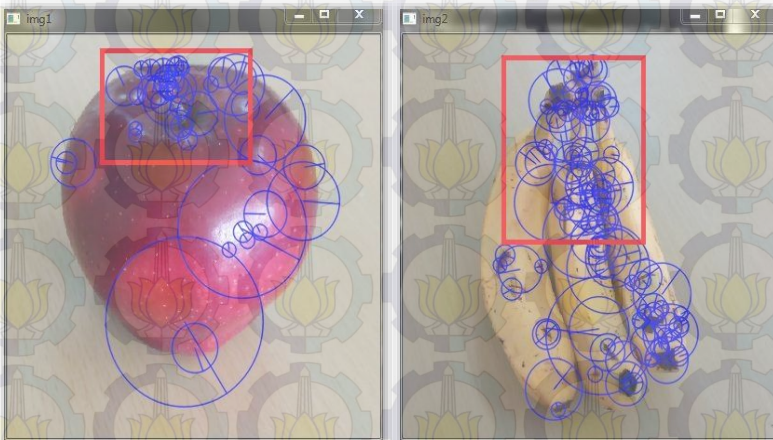
Tabel 5.8. Dokumentasi Hasil Pengujian 5

Uji Coba	Hasil Identifikasi	Waktu (detik)	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	Jeruk	11	11	Apel	9
2	Pisang	10	12	Apel	8
3	Apel	9	13	Apel	9
4	Apel	8	14	Apel	7
5	Apel	9	15	Apel	5
6	Jeruk	10	16	Apel	6
7	Jeruk	9	17	Jeruk	8
8	Apel	7	18	Jeruk	8
9	Apel	5	19	Apel	9
10	Apel	6	20	Apel	5
Tingkat kebenaran dari 20 uji coba = 70%					
Rata-rata waktu identifikasi = 7.9 detik					



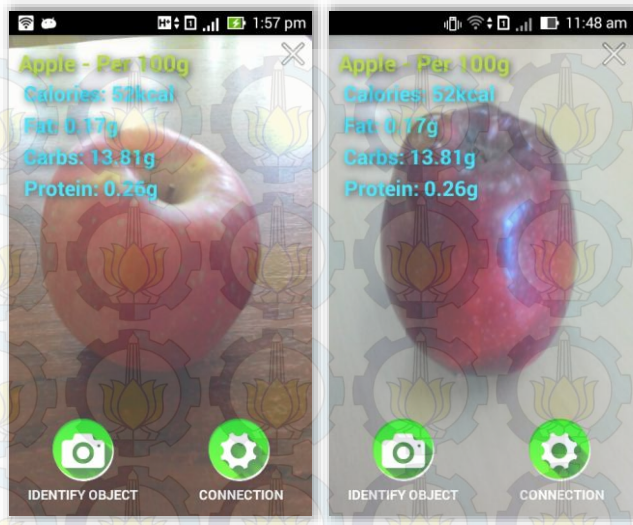
Gambar 5.7. Identifikasi Kemiripan Buah Apel dan Jeruk

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi buah apel benar sebanyak 14 dari 20 kali uji coba. Sedangkan 5 uji coba teridentifikasi sebagai buah jeruk dan 1 uji coba teridentifikasi sebagai buah pisang, hal ini dikarenakan *keypoint* pada buah apel memiliki kemiripan pada buah jeruk dan pisang ketika posisi pengambilan gambar untuk diidentifikasi pada uji coba 1, 2, 6, 7, 17, dan 18. Identifikasi kemiripan buah apel dan jeruk dapat dilihat pada Gambar 5.7. Sedangkan identifikasi kemiripan buah apel dan pisang dapat dilihat pada Gambar 5.8.



Gambar 5.8. Identifikasi Kemiripan Buah Apel dan Pisang

Karena fitur yang diambil hanya *keypoint* saja serta bukan fitur warna dan bentuk, maka buah jeruk dan pisang masih terdeteksi kemiripan dengan buah apel. Sehingga tingkat kebenaran dari 20 kali uji coba adalah 70%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 7.9 detik. Pengujian dilakukan terhadap 4 macam buah apel denganodus pengambilan gambar berbeda. Contoh uji coba pengujian aplikasi pada buah apel dapat dilihat pada Gambar 5.9.



Gambar 5.9. Contoh Uji Coba Pengujian Aplikasi pada Buah Apel

5.2.6. Pengujian Skenario 6 dan Evaluasi

Pada pengujian skenario 6 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Indomie. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.9.

Tabel 5.9. Skenario Pengujian 6

Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Produk Indomie
Kode	SP-06

Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada produk Indomie.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaan modus pengambilan gambar.
Data Input	Objek produk Indomie.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi produk Indomie sebanyak 20 kali. 2. Terdapat 4 macam posisi pengambilan gambar produk Indomie dengan masing – masing posisi dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi produk Indomie didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi produk Indomie memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi produk Indomie benar sebanyak 18 dari 20 kali uji coba

Tabel 5.10. Dokumentasi Hasil Pengujian 6

Uji Coba	Hasil Identifikasi	Waktu (detik)	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	Indomie	9	11	Indomie	9
2	Indomie	10	12	Indomie	8
3	Indomie	8	13	Indomie	10
4	Indomie	11	14	Indomie	9
5	Indomie	12	15	Indomie	11

Uji Coba	Hasil Identifikasi	Waktu (detik)	Uji Coba	Hasil Identifikasi	Waktu (detik)
6	Indomie	10	16	Indomie	9
7	Indomie	9	17	Indomie	8
8	Indomie	8	18	Indomie	10
9	Indomie	10	19	Indomie	9
10	Indomie	7	20	Indomie	8
Tingkat kebenaran dari 20 uji coba = 100%					
Rata-rata waktu identifikasi = 9.25 detik					



Gambar 5.10. Contoh Uji Coba Pengujian Aplikasi pada Produk Indomie

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi produk Indomie benar sebanyak 20 dari 20 kali uji coba. Sehingga tingkat kebenaran dari 20 kali uji coba adalah 100%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 9.25 detik. Pengujian dilakukan terhadap 4 macam posisi denganodus pengambilan gambar berbeda.

Contoh uji coba pengujian aplikasi pada produk Indomie dapat dilihat pada Gambar 5.10.

5.2.7. Pengujian Skenario 7 dan Evaluasi

Pada pengujian skenario 7 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Pringles. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.11.

Tabel 5.11. Skenario Pengujian 7

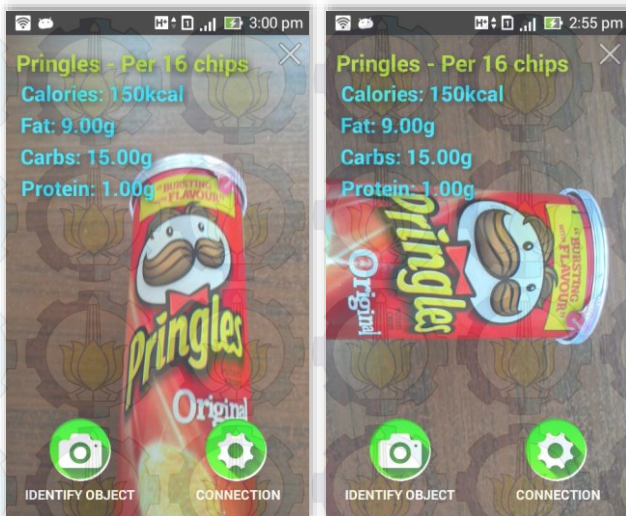
Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Produk Pringles
Kode	SP-07
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada produk Pringles.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaanodus pengambilan gambar.
Data Input	Objek produk Pringles.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi produk Pringles sebanyak 20 kali. 2. Terdapat 4 macam posisi pengambilan gambar produk Pringles dengan masing–masing posisi dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi produk Pringles

	didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi produk Pringles memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi produk Pringles benar sebanyak 20 dari 20 kali uji coba

Tabel 5.12. Dokumentasi Hasil Pengujian 7

Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	10	Pringles	11	Pringles	10
2	11	Pringles	12	Pringles	9
3	8	Pringles	13	Pringles	11
4	9	Pringles	14	Pringles	10
5	10	Pringles	15	Pringles	9
6	8	Pringles	16	Pringles	9
7	9	Pringles	17	Pringles	8
8	8	Pringles	18	Pringles	10
9	8	Pringles	19	Pringles	8
10	9	Pringles	20	Pringles	8
Tingkat kebenaran dari 20 uji coba = 100%					
Rata-rata waktu identifikasi = 9.1 detik					

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi produk Pringles benar sebanyak 20 dari 20 kali uji coba. Sehingga tingkat kebenaran dari 20 kali uji coba adalah 100%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 9.1 detik. Pengujian dilakukan terhadap 4 macam posisis dengan modus pengambilan gambar berbeda. Contoh uji coba pengujian aplikasi pada produk Pringles dapat dilihat pada Gambar 5.11.



Gambar 5.11. Contoh Uji Coba Pengujian Aplikasi pada Produk Pringles

5.2.8. Pengujian Skenario 8 dan Evaluasi

Pada pengujian skenario 8 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada buah lemon. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.13.

Tabel 5.13. Skenario Pengujian 8

Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Buah Lemon
-------------------------	---

Kode	SP-08
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada buah lemon.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaan modus pengambilan gambar.
Data Input	Objek buah lemon.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi buah lemon sebanyak 20 kali. 2. Terdapat 4 macam produk Pringles dengan masing-masing buah dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi buah lemon didokumentasikan kebenaran dan kesalahannya
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi buah lemon memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi buah lemon benar sebanyak 11 dari 20 kali uji coba

Tabel 5.14. Dokumentasi Hasil Pengujian 8

Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	7	Lemon	11	Lemon	8
2	6	Apel	12	Pisang	7
3	9	Pisang	13	Pisang	9
4	6	Lemon	14	Lemon	7
5	7	Pisang	15	Lemon	8
6	8	Lemon	16	Lemon	9
7	8	Lemon	17	Lemon	9

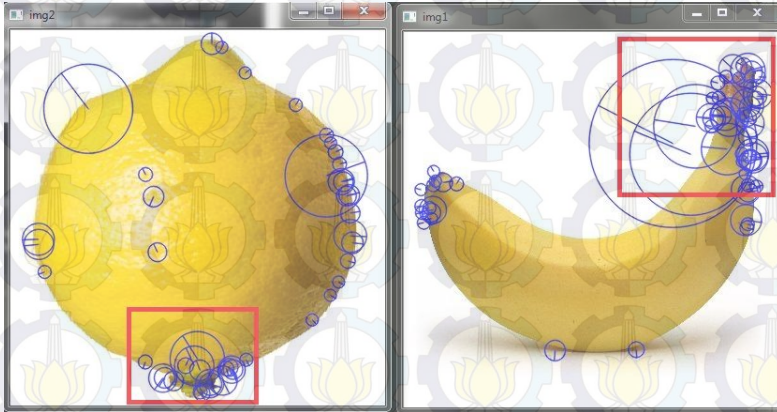
Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
8	6	Apel	18	Pisang	9
9	7	Apel	19	Lemon	8
10	8	Pisang	20	Lemon	9
Tingkat kebenaran dari 20 uji coba = 55%					
Rata-rata waktu identifikasi = 7.75 detik					



Gambar 5.12. Contoh Uji Coba Pengujian Aplikasi pada Buah Lemon

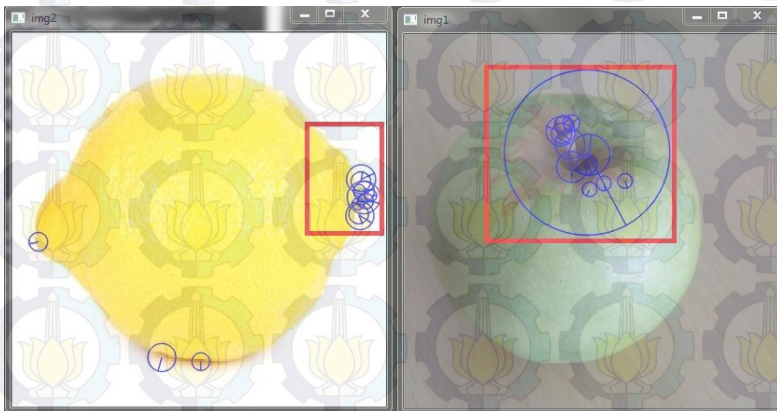
Gambar 5.12. adalah contoh uji coba pengujian aplikasi pada buah lemon. Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi buah lemon benar sebanyak 11 dari 20 kali uji coba. Sedangkan 3 uji coba teridentifikasi sebagai buah apel dan 6 uji coba teridentifikasi sebagai buah pisang, hal ini dikarenakan *keypoint* pada buah lemon memiliki kemiripan pada buah apel dan pisang. Faktor lain yang mempengaruhi adalah posisi modus kamera pada saat pengambilan gambar. Sehingga

diperoleh rata-rata prosentase kebenaran dalam mengidentifikasi objek sebesar 55%. Rata-rata waktu yang dibutuhkan untuk identifikasi selama 7.75 detik. Identifikasi kemiripan buah lemon dan pisang dapat dilihat pada Gambar 5.13.



Gambar 5.13. Identifikasi Kemiripan Buah Lemon dan Pisang

Sedangkan identifikasi kemiripan buah lemon dan apel dapat dilihat pada Gambar 5.14.



Gambar 5.14. Identifikasi Kemiripan Buah Lemon dan Apel

5.2.9. Pengujian Skenario 9 dan Evaluasi

Pada pengujian skenario 9 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Pocky. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.15.

Tabel 5.15. Skenario Pengujian 9

Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Produk Pocky
Kode	SP-09
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada produk Pocky.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaan modus pengambilan gambar.
Data Input	Objek produk Pocky.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi produk Pocky sebanyak 20 kali. 2. Terdapat 4 macam posisi pengambilan gambar produk Pocky dengan masing-masing posisi dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi produk Pocky didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.

Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi produk Pocky memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi produk Pocky benar sebanyak 20 dari 20 kali uji coba

Tabel 5.16. Dokumentasi Hasil Pengujian 9

Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	9	Pocky	11	Pocky	9
2	10	Pocky	12	Pocky	10
3	11	Pocky	13	Pocky	10
4	9	Pocky	14	Pocky	9
5	9	Pocky	15	Pocky	10
6	10	Pocky	16	Pocky	11
7	9	Pocky	17	Pocky	11
8	9	Pocky	18	Pocky	10
9	10	Pocky	19	Pocky	9
10	11	Pocky	20	Pocky	10
Tingkat kebenaran dari 20 uji coba = 100 %					
Rata-rata waktu identifikasi = 9.8 detik					

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi produk Pocky benar sebanyak 20 dari 20 kali uji coba. Sehingga tingkat kebenaran dari 20 kali uji coba adalah 100%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 9.8 detik. Pengujian dilakukan terhadap 4 macam posisi dengan modus pengambilan gambar berbeda. Contoh uji coba pengujian aplikasi pada produk Pocky dapat dilihat pada Gambar 5.15.



Gambar 5.15. Contoh Uji Coba Pengujian Aplikasi pada Produk Pocky

5.2.10. Pengujian Skenario 10 dan Evaluasi

Pada pengujian skenario 10 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Oreo. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.17.

Tabel 5.17. Skenario Pengujian 10

Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Produk Oreo
-------------------------	--

Kode	SP-10
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada produk Oreo.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaan modus pengambilan gambar.
Data Input	Objek produk Oreo.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi produk Oreo sebanyak 20 kali. 2. Terdapat 4 macam posisi pengambilan gambar produk Oreo dengan masing-masing posisi dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi produk Oreo didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi produk Oreo memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi produk Oreo benar sebanyak 20 dari 20 kali uji coba

Tabel 5.18. Dokumentasi Hasil Pengujian 10

Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	8	Oreo	11	Oreo	10
2	7	Oreo	12	Oreo	9
3	7	Oreo	13	Oreo	8
4	9	Oreo	14	Oreo	10

Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
5	9	Oreo	15	Oreo	9
6	8	Oreo	16	Oreo	9
7	7	Oreo	17	Oreo	8
8	9	Oreo	18	Oreo	11
9	9	Oreo	19	Oreo	10
10	11	Oreo	20	Oreo	10
Tingkat kebenaran dari 20 uji coba = 100 %					
Rata-rata waktu identifikasi = 8.9 detik					



Gambar 5.16. Contoh Uji Coba Pengujian Aplikasi pada Produk Oreo

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi produk Oreo benar sebanyak 20 dari 20 kali uji coba. Sehingga tingkat kebenaran dari 20 kali uji coba adalah 100%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 8.9 detik. Pengujian dilakukan terhadap

4 macam posisi denganodus pengambilan gambar berbeda. Contoh uji coba pengujian aplikasi pada produk Oreo dapat dilihat pada Gambar 5.16.

5.2.11. Pengujian Skenario 11 dan Evaluasi

Pada pengujian skenario 11 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Snickers. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.19.

Tabel 5.19. Skenario Pengujian 11

Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Produk Snickers
Kode	SP-11
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada produk Snickers.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaanodus pengambilan gambar.
Data Input	Objek produk Snickers.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi produk Snickers sebanyak 20 kali. 2. Terdapat 4 macam posisi pengambilan gambar produk Snickers dengan masing-masing posisi dilakukan 5 kali uji coba.

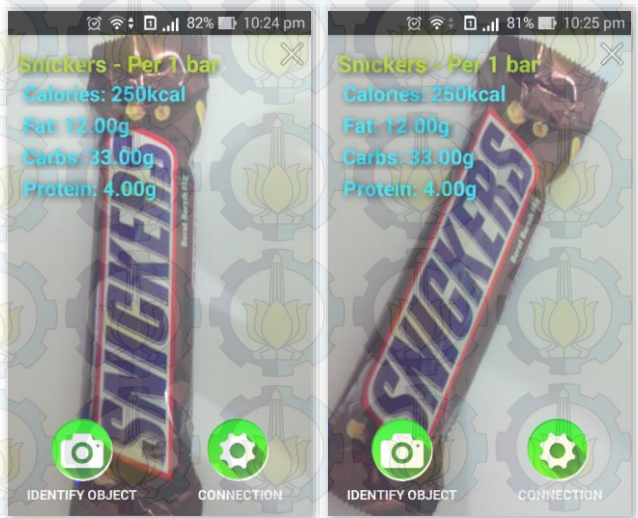
	3. Hasil uji coba identifikasi nama dan nutrisi produk Snickers didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi produk Snickers memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi produk Snickers benar sebanyak 20 dari 20 kali uji coba

Tabel 5.20. Dokumentasi Hasil Pengujian 11

Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	11	Snickers	11	Snickers	9
2	10	Snickers	12	Snickers	10
3	12	Snickers	13	Snickers	12
4	11	Snickers	14	Snickers	12
5	11	Snickers	15	Snickers	10
6	12	Snickers	16	Snickers	11
7	10	Snickers	17	Snickers	10
8	13	Snickers	18	Snickers	10
9	10	Snickers	19	Snickers	10
10	12	Snickers	20	Snickers	11
Tingkat kebenaran dari 20 uji coba = 100 %					
Rata-rata waktu identifikasi = 10.85 detik					

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi produk Snickers benar sebanyak 20 dari 20 kali uji coba. Sehingga tingkat kebenaran dari 20 kali uji coba adalah 100%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 10.85 detik. Pengujian dilakukan terhadap 4 macam posisi dengan modus pengambilan gambar

berbeda. Contoh uji coba pengujian aplikasi pada produk Snickers dapat dilihat pada Gambar 5.17.



Gambar 5.17. Contoh Uji Coba Pengujian Aplikasi pada Produk Snickers

5.2.12. Pengujian Skenario 12 dan Evaluasi

Pada pengujian skenario 12 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Tango. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.21.

Tabel 5.21. Skenario Pengujian 12

Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan
-------------------------	---

	Kandungan Nutrisi pada Produk Tango
Kode	SP-12
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada produk Tango.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaan modus pengambilan gambar.
Data Input	Objek produk Tango.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi produk Tango sebanyak 20 kali. 2. Terdapat 4 macam posisi pengambilan gambar produk Tango dengan masing-masing posisi dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi produk Tango didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi produk Tango memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi produk Tango benar sebanyak 20 dari 20 kali uji coba

Tabel 5.22. Dokumentasi Hasil Pengujian 12

Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	11	Tango	11	Tango	11
2	13	Tango	12	Tango	11

Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
3	10	Tango	13	Tango	10
4	10	Tango	14	Tango	10
5	13	Tango	15	Tango	11
6	12	Tango	16	Tango	12
7	11	Tango	17	Tango	13
8	13	Tango	18	Tango	10
9	11	Tango	19	Tango	11
10	12	Tango	20	Tango	12
Tingkat kebenaran dari 20 uji coba = 100 %					
Rata-rata waktu identifikasi = 11.35 detik					



Gambar 5.18. Contoh Uji Coba Pengujian Aplikasi pada Produk Tango

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi produk Tango benar sebanyak 20 dari 20 kali uji coba. Sehingga tingkat kebenaran dari 20 kali uji coba

adalah 100%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 11.35 detik. Pengujian dilakukan terhadap 4 macam posisi dengan modus pengambilan gambar berbeda. Contoh uji coba pengujian aplikasi pada produk Snickers dapat dilihat pada Gambar 5.18.

5.2.13. Pengujian Skenario 13 dan Evaluasi

Pengujian skenario 13 ini merupakan kalkulasi kebenaran aplikasi dalam mengidentifikasi nama dan nutrisi. Kalkulasi kebenaran aplikasi adalah hasil dari keseluruhan uji coba dari pengujian 3 hingga 12. Hasil pengujian dapat dilihat pada Tabel 5.23.

Tabel 5.23. Hasil Pengujian 13

		Target Kelas									
		Pisang	Jeruk	Apel	Indomie	Pringles	Lemon	Pocky	Oreo	Snickers	Tango
Hasil Uji Coba	Pisang	20	2	1			6				
	Jeruk		18	5							
	Apel			14			3				
	Indomie				20						
	Pringles					20					
	Lemon						11				
	Pocky							20			
	Oreo								20		
	Snickers									20	
	Tango										20

Sedangkan hasil prosentase kebenaran identifikasi objek makanan atau produk makanan dapat dilihat pada Tabel 5.24. Hasil kalkulasi prosentase kebenaran merupakan rata-rata dari pengujian skenario 3 hingga pengujian skenario 12.

Tabel 5.24. Hasil Kalkulasi Prosentase Kebenaran

Nama Objek	Jumlah Benar	Prosentase Benar
Pisang	20	100%
Jeruk	18	90%
Apel	14	70%
Indomie	20	100%
Pringles	20	100%
Lemon	11	55%
Pocky	20	100%
Oreo	20	100%
Snickers	20	100%
Tango	20	100%
RATA – RATA		92%

Tabel 5.25. Hasil Kalkulasi Waktu Identifikasi

Nama Objek	Rata-Rata Waktu Identifikasi (detik)
Pisang	9.4
Jeruk	8.65
Apel	7.9
Indomie	9.25
Pringles	9.1
Lemon	7.75
Pocky	9.8
Oreo	8.9
Snickers	10.85
Tango	11.35
RATA–RATA KESELURUHAN	9.295

Pada pengujian skenario 3 hingga 12 didapatkan rata-rata prosentase kebenaran aplikasi dalam mengidentifikasi objek makanan atau produk makanan sebesar 92%. Identifikasi kebenaran pada buah apel dan lemon memiliki prosentase yang rendah dibandingkan dengan 8 objek yang lainnya. Hal ini dikarenakan fitur yang diambil hanya *keypoint* saja serta bukan

fitur warna dan bentuk. Faktor lain yang mempengaruhi yaitu modus kamera pada saat posisi pengambilan gambar untuk diidentifikasi. Sedangkan lamanya waktu yang diperlukan untuk mengidentifikasi nama dan kandungan nutrisi tergantung pada kompleksitas fitur *keypoint* pada gambar. Hasil kalkulasi waktu identifikasi dapat dilihat pada Tabel 5.25.

5.2.14. Pengujian Skenario 14 dan Evaluasi

Pada pengujian skenario 14 ini dilakukan terhadap pengguna secara langsung yaitu dengan melibatkan Mahasiswa dan kalangan masyarakat sekitar ITS sebagai pengguna langsung aplikasi. Dalam pengujian, dilakukan penyebaran kuesioner sebagai timbal balik dari pengguna ke pengembang aplikasi.

5.2.14.1. Daftar Responden Sebagai Penguji

Pada subbab ini ditunjukkan daftar Responden yang bertindak sebagai penguji coba aplikasi yang telah dibangun. Daftar Responden sebagai penguji dapat dilihat pada Tabel 5.26.

Tabel 5.26. Daftar Responden Sebagai Penguji

Nama	Rentang Umur (tahun)	Jenis Kelamin
Responden 1	21 – 30	Laki-laki
Responden 2	21 – 30	Laki-laki
Responden 3	21 – 30	Perempuan
Responden 4	21 – 30	Laki-laki
Responden 5	21 – 30	Perempuan
Responden 6	21 – 30	Perempuan
Responden 7	21 – 30	Perempuan
Responden 8	31 – 40	Perempuan
Responden 9	13 – 20	Perempuan
Responden 10	21 – 30	Laki-laki

5.2.14.2. Hasil Uji Coba Kepada Responden Sebagai Penguji

Uji coba kepada pengguna sebagai penguji menggunakan metode penyebaran kuesioner sebagai timbal balik dari pengguna ke pengembang setelah menggunakan aplikasi. Hasil uji coba kepada pengguna sebagai penguji dapat dilihat pada Tabel 5.27.

Tabel 5.27. Hasil Uji Coba Kepada Responden Sebagai Penguji

Kode	Indikator	Rata – Rata Nilai (Rentang 1-5)
UC-1	Pengguna selalu memperhatikan nutrisi makanan yang dikonsumsi sehari-hari	2.4
UC-2	Kemudahan aplikasi ketika digunakan untuk mendapatkan informasi nutrisi	3.7
UC-3	Ketertarikan pengguna terhadap aplikasi untuk mengontrol konsumsi nutrisi sehari-hari	3.9
UC-4	Kebenaran aplikasi dalam mendeteksi nama dan nutrisi objek makanan	3.3
UC-5	Kemudahan menavigasikan icon aplikasi dalam penggunaan aplikasi	4.2
RATA – RATA TOTAL		3.5

Sedangkan detail pengujian pada setiap aspek indikator kuesioner yang disebar kepada pengguna sebagai penguji langsung dapat dilihat pada Tabel 5.28.

Tabel 5.28. Detil Pengujian pada Setiap Aspek Indikator

Nama Pengguna	Aspek Indikator				
	UC-1	UC-2	UC-3	UC-4	UC-5
Responden 1	1	4	4	4	4
Responden 2	3	4	4	4	4

Nama Pengguna	Aspek Indikator				
	UC-1	UC-2	UC-3	UC-4	UC-5
Responden 3	3	3	3	2	5
Responden 4	2	5	5	4	5
Responden 5	4	4	4	3	4
Responden 6	4	3	3	3	4
Responden 7	4	4	4	4	5
Responden 8	1	3	4	3	3
Responden 9	1	3	4	3	4
Responden 10	2	4	4	3	4
TOTAL	22	33	35	30	38

BAB VI

KESIMPULAN DAN SARAN

Pada bab ini diberikan kesimpulan yang diambil selama pengerjaan tugas akhir serta saran-saran tentang pengembangan yang dapat dilakukan terhadap tugas akhir ini di masa yang akan datang.

6.1. Kesimpulan

Dari proses pengerjaan selama perancangan, implementasi, dan proses pengujian aplikasi yang dilakukan, dapat diambil kesimpulan sebagai berikut.

1. Aplikasi dapat mengidentifikasi jenis makanan atau produk makanan dari masukan berupa gambar menggunakan metode klasifikasi gambar, sehingga bisa didapatkan informasi berupa nama dari objek makanan atau produk makanan yang diidentifikasi.
2. Metode yang digunakan untuk ekstraksi fitur *keypoint* gambar adalah SURF (*Speeded-Up Robust Feature*) yang sudah disediakan oleh pustaka OpenCV. Setelah mendapatkan fitur *keypoint*, dilakukan perhitungan jarak terdekat menggunakan *Euclidian Distance* untuk klasifikasi objek gambar. Apabila jarak yang dihitung semakin kecil maka kemiripan semakin besar.
3. Aplikasi dapat mengidentifikasi kandungan nutrisi berupa kalori, karbohidrat, protein dan lemak di dalam makanan atau produk makanan menggunakan basis data pada *webservice Fatsecret*. Masukan untuk *webservice Fatsecret* adalah hasil klasifikasi gambar di *webservice* yaitu berupa nama makanan atau produk makanan. Nama makanan atau produk makanan kemudian dikirimkan ke pencarian kandungan nutrisi pada basis data *Fatsecret*. Keluaran

pencarian kandungan nutrisi pada basis data *Fatsecret* berupa rangkuman nutrisi per takaran saji.

4. Pengujian kebenaran aplikasi dalam mengidentifikasi nama dan informasi nutrisi yang terkandung memiliki rata-rata prosentase kebenaran sebesar 92% dengan rata-rata waktu 9.295 detik.
5. Identifikasi buah apel dan lemon memiliki prosentase akurasi rendah dibandingkan dengan objek yang lainnya. Hal ini dikarenakan fitur yang diambil hanya *keypoint* saja, serta bukan fitur warna dan bentuk. Faktor lain yang mempengaruhi yaitu posisi modus kamera pada saat pengambilan gambar objek untuk diidentifikasi.
6. Aplikasi dapat menampilkan informasi nutrisi berupa kalori, karbohidrat, protein dan lemak per takaran saji menggunakan teknologi *augmented reality* teks berbasis *android* sehingga lebih menarik.

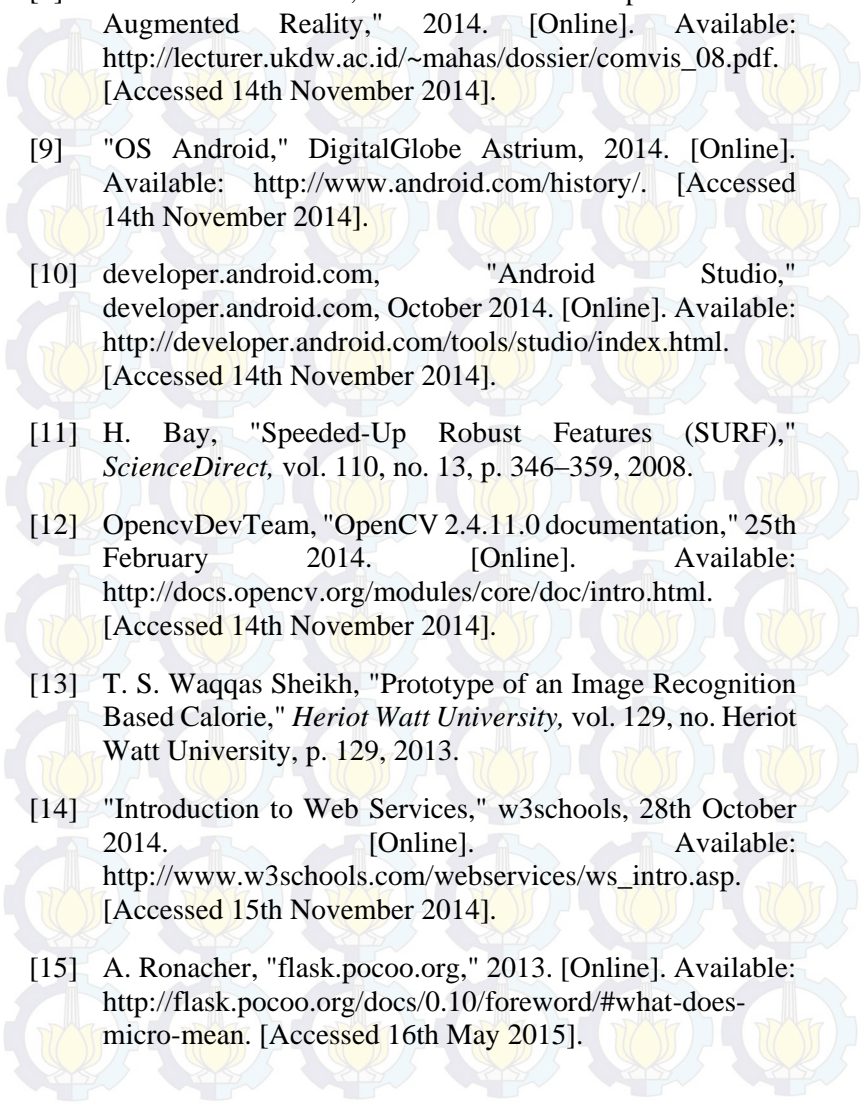
6.2. Saran

Berikut saran-saran untuk pengembangan dan perbaikan sistem di masa yang akan datang. Di antaranya adalah sebagai berikut:

1. Untuk memberikan kesan yang lebih informatif perlu ditampilkan suatu grafik yang dinamis berdasarkan masukan informasi kandungan nutrisi dari *Fatsecret* API. Sampai saat ini pengembang belum menemukan cara bagaimana menampilkan grafik secara dinamis pada permukaan layar modus kamera *android*.
2. Diperlukan suatu SDK (*Software Development Kit*) khusus untuk menampilkan *augmented reality* teks tanpa menggunakan proses *rendering* dari teks menjadi gambar sehingga bisa dinamis. Untuk saat ini pengembang menampilkan informasi nutrisi dengan membangun secara tradisional pada layar permukaan modus kamera *android*.

DAFTAR PUSTAKA

- [1] World Health Organization, "Obesity and Overweight Fact Sheet," World Health Organization, 2nd March 2013. [Online]. Available: <http://www.who.int/mediacentre/factsheets/fs311/en/>. [Accessed 6th November 2014].
- [2] S. Langley-Evans, "Human Nutrition and Dietetics," *Journal of Human Nutrition and Dietetics*, vol. 79, p. 48, Juni 2013.
- [3] L. A. Times, "Diet," Los Angeles Times, 21 October 2014. [Online]. Available: <http://articles.latimes.com/keyword/diet>. [Accessed 7th November 2014].
- [4] Yuku, "Nutrition Facts Apps Store," Nutrition Facts, 30th September 2014. [Online]. Available: <https://play.google.com/store/apps/details?id=yuku.usdanutlite&hl=en>. [Accessed 7th November 2014].
- [5] H. A. N. M. A. Muhammad Zulfikar Bayu, "Nutritional Information Visualization Using Mobile Augmented Reality Technology," *ScienceDirect*, vol. 11, no. The 4th International Conference on Electrical Engineering and Informatics (ICEEI 2013), pp. 396-402, 2013.
- [6] "Nutrition Definition," 2014. [Online]. Available: <http://www.who.int/topics/nutrition/en/>. [Accessed 14th November 2014].
- [7] "Calorie Counting - A Guide to Calories & Weight Control," 2014. [Online]. Available: <http://www.acaloriecounter.com/calorie-counting.php>. [Accessed 14th November 2014].

- 
- [8] A. W. Mahastama, "Pemanfaatan Computer Vision: Augmented Reality," 2014. [Online]. Available: http://lecturer.ukdw.ac.id/~mahas/dossier/comvis_08.pdf. [Accessed 14th November 2014].
- [9] "OS Android," DigitalGlobe Astrium, 2014. [Online]. Available: <http://www.android.com/history/>. [Accessed 14th November 2014].
- [10] developer.android.com, "Android Studio," developer.android.com, October 2014. [Online]. Available: <http://developer.android.com/tools/studio/index.html>. [Accessed 14th November 2014].
- [11] H. Bay, "Speeded-Up Robust Features (SURF)," *ScienceDirect*, vol. 110, no. 13, p. 346–359, 2008.
- [12] OpencvDevTeam, "OpenCV 2.4.11.0 documentation," 25th February 2014. [Online]. Available: <http://docs.opencv.org/modules/core/doc/intro.html>. [Accessed 14th November 2014].
- [13] T. S. Waqqas Sheikh, "Prototype of an Image Recognition Based Calorie," *Heriot Watt University*, vol. 129, no. Heriot Watt University, p. 129, 2013.
- [14] "Introduction to Web Services," w3schools, 28th October 2014. [Online]. Available: http://www.w3schools.com/webservices/ws_intro.asp. [Accessed 15th November 2014].
- [15] A. Ronacher, "flask.pocoo.org," 2013. [Online]. Available: <http://flask.pocoo.org/docs/0.10/foreword/#what-does-micro-mean>. [Accessed 16th May 2015].

BAB VI

KESIMPULAN DAN SARAN

Pada bab ini diberikan kesimpulan yang diambil selama pengerjaan tugas akhir serta saran-saran tentang pengembangan yang dapat dilakukan terhadap tugas akhir ini di masa yang akan datang.

6.1. Kesimpulan

Dari proses pengerjaan selama perancangan, implementasi, dan proses pengujian aplikasi yang dilakukan, dapat diambil kesimpulan sebagai berikut.

1. Aplikasi dapat mengidentifikasi jenis makanan atau produk makanan dari masukan berupa gambar menggunakan metode klasifikasi gambar, sehingga bisa didapatkan informasi berupa nama dari objek makanan atau produk makanan yang diidentifikasi.
2. Metode yang digunakan untuk ekstraksi fitur *keypoint* gambar adalah SURF (*Speeded-Up Robust Feature*) yang sudah disediakan oleh pustaka OpenCV. Setelah mendapatkan fitur *keypoint*, dilakukan perhitungan jarak terdekat menggunakan *Euclidian Distance* untuk klasifikasi objek gambar. Apabila jarak yang dihitung semakin kecil maka kemiripan semakin besar.
3. Aplikasi dapat mengidentifikasi kandungan nutrisi berupa kalori, karbohidrat, protein dan lemak di dalam makanan atau produk makanan menggunakan basis data pada *webservice Fatsecret*. Masukan untuk *webservice Fatsecret* adalah hasil klasifikasi gambar di *webservice* yaitu berupa nama makanan atau produk makanan. Nama makanan atau produk makanan kemudian dikirimkan ke pencarian kandungan nutrisi pada basis data *Fatsecret*. Keluaran

pencarian kandungan nutrisi pada basis data *Fatsecret* berupa rangkuman nutrisi per takaran saji.

4. Pengujian kebenaran aplikasi dalam mengidentifikasi nama dan informasi nutrisi yang terkandung memiliki rata-rata prosentase kebenaran sebesar 92% dengan rata-rata waktu 9.295 detik.
5. Identifikasi buah apel dan lemon memiliki prosentase akurasi rendah dibandingkan dengan objek yang lainnya. Hal ini dikarenakan fitur yang diambil hanya *keypoint* saja, serta bukan fitur warna dan bentuk. Faktor lain yang mempengaruhi yaitu posisi modus kamera pada saat pengambilan gambar objek untuk diidentifikasi.
6. Aplikasi dapat menampilkan informasi nutrisi berupa kalori, karbohidrat, protein dan lemak per takaran saji menggunakan teknologi *augmented reality* teks berbasis *android* sehingga lebih menarik.

6.2. Saran

Berikut saran-saran untuk pengembangan dan perbaikan sistem di masa yang akan datang. Di antaranya adalah sebagai berikut:

1. Untuk memberikan kesan yang lebih informatif perlu ditampilkan suatu grafik yang dinamis berdasarkan masukan informasi kandungan nutrisi dari *Fatsecret* API. Sampai saat ini pengembang belum menemukan cara bagaimana menampilkan grafik secara dinamis pada permukaan layar modus kamera *android*.
2. Diperlukan suatu SDK (*Software Development Kit*) khusus untuk menampilkan *augmented reality* teks tanpa menggunakan proses *rendering* dari teks menjadi gambar sehingga bisa dinamis. Untuk saat ini pengembang menampilkan informasi nutrisi dengan membangun secara tradisional pada layar permukaan modus kamera *android*.

LAMPIRAN (A) KODE SUMBER

```
public class MainActivity extends ActionBarActivity
implements View.OnClickListener {

    private Camera mCamera = null;
    private CameraView mCameraView = null;
    private String filepath;
    private TextView objects, calories, fat, carbs,
    protein;
    SharedPreferences sp;

    private Camera.PictureCallback mPicture = new
    Camera.PictureCallback() {

        @Override
        public void onPictureTaken(byte[] data,
        Camera camera) {

            File pictureFile = new
            File(Environment.getExternalStorageDirectory(),
            "testTA.jpg");

            filepath =
            pictureFile.getAbsolutePath();
            try {
                FileOutputStream fos = new
                FileOutputStream(pictureFile);
                fos.write(data);
                fos.close();
            } catch (Exception e) {
                e.printStackTrace();
            }
            new UploadImage(filepath).execute();
        }
    };

    @Override
    protected void onCreate(Bundle
    savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        sp =
        PreferenceManager.getDefaultSharedPreferences(this);
        PreferenceManager.setDefaultValues(this,
```



```

R.xml.preference_screen, false);

        objects = (TextView)
findViewById(R.id.objectTxt);
        calories = (TextView)
findViewById(R.id.caloriesTxt);
        fat = (TextView) findViewById(R.id.fatTxt);
        carbs = (TextView)
findViewById(R.id.carbsTxt);
        protein = (TextView)
findViewById(R.id.proteinTxt);

        ImageView imgPic = (ImageView)
findViewById(R.id.btn_identify);
        imgPic.setOnClickListener(this);

        ImageView imgSet = (ImageView)
findViewById(R.id.btn_setting);
        imgSet.setOnClickListener(this);

        ImageButton imgClose =
        (ImageButton) findViewById(R.id.imgClose);
        imgClose.setOnClickListener(this);
    }

```

Kode Sumber 7.1. Kode MainActivity.java (Bagian 1)

```

@Override
protected void onPause() {
    super.onPause();
    try{
        mCamera.stopPreview();

        mCameraView.getHolder().removeCallback(mCameraView);
        mCamera.release();
        mCamera = null;
    }catch (Exception e){
        e.printStackTrace();
    }
}

@Override
protected void onResume() {
    super.onResume();
    getCameraInstance();
}

```

```

}
@Override
public void onClick(View v) {
    if (v.getId() == R.id.btn_identify) {
        mCamera.takePicture(null, null, mPicture);
    } else if (v.getId() == R.id.btn_setting) {
        Intent intent = new Intent(this,
        SettingActivity.class);
        this.startActivity(intent);
    } else if (v.getId() == R.id.imgClose) {
        System.exit(1);
    }
}
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu_main,
    menu);
    return true;
}

private void getCameraInstance() {
    try {
        mCamera = Camera.open();
    } catch (Exception e) {
        Log.d("ERROR", "Failed to get camera: " +
        e.getMessage());
    }
    if (mCamera != null) {
        Camera.Parameters params =
        mCamera.getParameters();
        params.setFocusMode(Camera.Parameters.FOCUS_MODE_AUTO);
        params.setPictureSize(400, 400);
        params.setRotation(90);
        mCamera.setParameters(params);
        mCameraView = new CameraView(this, mCamera);
        FrameLayout camera_view =
        (FrameLayout) findViewById(R.id.camera_view);
        camera_view.addView(mCameraView);
    }
}
}

```

Kode Sumber 7.2. Kode MainActivity.java (Bagian 2)

```

private void updateInfo(String hasil){
    String[] split0 = hasil.split("#");
    String[] split1 = split0[1].split("-");
    String[] split2 = split1[1].split("\\|");
    objects.setText(split0[0] + " - " + split1[0]);
    calories.setText(split2[0]);
    fat.setText(split2[1]);
    carbs.setText(split2[2]);
    protein.setText(split2[3]);
    Handler h = new Handler();
    h.postDelayed(new Runnable() {
        public void run() {
            //change your text here
            objects.setText("");
            calories.setText("");
            fat.setText("");
            carbs.setText("");
            protein.setText("");
        }
    }, 10000);
}
@Override
public boolean onOptionsItemSelected(MenuItem item)
{
    int id = item.getItemId();
    if (id == R.id.action_settings) {
        Intent intent = new Intent(this,
        SettingActivity.class);
        this.startActivity(intent);
        return true;
    }
    return super.onOptionsItemSelected(item);
}

```

Kode Sumber 7.3. Kode MainActivity.java (Bagian 3)

```

@Override
protected String doInBackground(String... params) {
    String urlAddr =
    sp.getString(SettingActivity.KEY_PREF_SERVER_ADDRESS
    , "");
    File file = new File(namaImage);
    HttpURLConnection httpConn = null;
    DataOutputStream dos = null;

```

```

String lineEnd = "\r\n";
String twoHyphens = "--";
String boundary = "*****";
int bytesRead, bytesAvailable, maxBufferSize;
byte[] buffer;

try {
    FileInputStream fus = new
FileInputStream(file);
    URL url = new URL(urlAddr);
    httpConn = (URLConnection)
url.openConnection();
    httpConn.setDoInput(true);
    httpConn.setDoOutput(true);
    httpConn.setUseCaches(false);
    httpConn.setRequestMethod("POST");
    httpConn.setRequestProperty("Connection",
"Keep-Alive");
    httpConn.setRequestProperty("ENCTYPE",
"multipart/form-data");
    httpConn.setRequestProperty("Content-Type",
"multipart/form-data;boundary=" + boundary);

    dos = new
DataOutputStream(httpConn.getOutputStream());
    dos.writeBytes(twoHyphens + boundary +
lineEnd);
    dos.writeBytes("Content-Disposition: form-
data; name=\"file\"; filename=\"" + "testTA.jpg\" +
lineEnd);
    dos.writeBytes(lineEnd);

    bytesAvailable = fus.available();
    maxBufferSize = 1024 * 1024;
    buffer = new byte[bytesAvailable];
    bytesRead = fus.read(buffer, 0,
bytesAvailable);
    while (bytesRead > 0) {
        dos.write(buffer, 0, bytesAvailable);
        bytesAvailable = fus.available();
        bytesAvailable =
Math.min(bytesAvailable, maxBufferSize);
        bytesRead = fus.read(buffer, 0,

```



```

bytesAvailable);
    }
    dos.writeBytes(lineEnd);
    dos.writeBytes(twoHyphens + boundary +
twoHyphens + lineEnd);

    fus.close();
    dos.flush();
    dos.close();

    int serverResponseCode =
httpConn.getResponseCode();
    String serverResponseMessage =
httpConn.getResponseMessage();
    InputStream is = httpConn.getInputStream();
    final String hasil =
convertStreamToString(is);
    Log.i("hasil = ", hasil);

    runOnUiThread(new Runnable() {
        @Override
        public void run() {
            updateInfo(hasil);
        }
    });

    Log.i("uploadFile", "HTTP Response is : " +
serverResponseMessage + ": " + serverResponseCode);
} catch (Exception e) {
    e.printStackTrace();
}
return null;
}
@Override
protected void onProgressUpdate(String...
values) {
    super.onProgressUpdate(values);
    if (values[0].matches("0")) {
        Toast.makeText(getApplicationContext(),
"Transfer image failed!!",
Toast.LENGTH_LONG).show();
    } else {
        Toast.makeText(getApplicationContext(),
"Transfer image is completed!!",

```

```

Toast.LENGTH_LONG).show();
    }
}

@Override
protected void onPostExecute(String s) {
    super.onPostExecute(s);
    pDialog.dismiss();
    mCamera.startPreview();
}
}

```

Kode Sumber 7.4. Kode MainActivity.java (Bagian 4)

```

public static int getExifOrientation(String
filepath){
    int degree = 0;
    ExifInterface exif = null;

    try {
        exif = new ExifInterface(filepath);
    } catch (Exception ex){
        ex.printStackTrace();
    }
    if (exif != null) {
        int orientation =
exif.getAttributeInt(ExifInterface.TAG_ORIENTATION,
-1);
        if (orientation != -1) {
            switch (orientation) {
                case
ExifInterface.ORIENTATION_ROTATE_90:
                    degree = 90;
                    break;
                case
ExifInterface.ORIENTATION_ROTATE_180:
                    degree = 180;
                    break;
                case
ExifInterface.ORIENTATION_ROTATE_270:
                    degree = 270;
                    break;
            }
        }
    }
}

```

```

    }
    return degree;
}

private static String
convertStreamToString(InputStream is) {

    BufferedReader reader = new
    BufferedReader(new InputStreamReader(is));
    StringBuilder sb = new StringBuilder();

    String line = null;
    try {
        while ((line = reader.readLine()) !=
null) {
            sb.append((line + "\n"));
        }
    } catch (IOException e) {
        e.printStackTrace();
    } finally {
        try {
            is.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
    return sb.toString();
}
}

```

Kode Sumber 7.5. Kode MainActivity.java (Bagian 5)

```

public class CameraView extends SurfaceView
implements SurfaceHolder.Callback{

    private SurfaceHolder mHolder;
    private Camera mCamera;

    public CameraView(Context context, Camera
camera){
        super(context);
        mCamera = camera;
        mHolder = getHolder();
        mHolder.addCallback(this);
    }
}

```

```

mHolder.setType(SurfaceHolder.SURFACE_TYPE_NORMAL);
    }

    @Override
    public void surfaceCreated(SurfaceHolder
surfaceHolder) {
        try{
            mCamera.setPreviewDisplay(surfaceHolder);
            mCamera.startPreview();
            mCamera.setDisplayOrientation(90);
        } catch (IOException e) {
            Log.d("ERROR", "Camera error on
surfaceCreated " + e.getMessage());
        }
    }

    @Override
    public void surfaceChanged(SurfaceHolder
surfaceHolder, int i, int i2, int i3) {
        if(mHolder.getSurface() == null)
            return;

        try{
            mCamera.stopPreview();
        } catch (Exception e){
        }

        try{
            mCamera.setPreviewDisplay(mHolder);
            mCamera.startPreview();
        } catch (IOException e) {
            Log.d("ERROR", "Camera error on
surfaceChanged " + e.getMessage());
        }
    }

    @Override
    public void surfaceDestroyed(SurfaceHolder
surfaceHolder) {
    }
}

```

Code Sumber 7.6. Kode CameraView.java


```

<FrameLayout
xmlns:android="http://schemas.android.com/apk/res/an
droid"
xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <FrameLayout
        android:id="@+id/camera_view"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:focusable="true"
        android:focusableInTouchMode="true" >

    </FrameLayout>

    <TextView
        style="@style/ButtonText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"

        android:textColor="@android:color/holo_green_light"
        android:textSize="24dp"
        android:textStyle="bold"
        android:layout_marginLeft="10dp"
        android:paddingTop="20px"
        android:id="@+id/objectTxt"
        android:layout_gravity="left|top" />

    <TextView
        style="@style/ButtonText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="40px"

        android:textColor="@android:color/holo_blue_bright"
        android:textSize="22dp"
        android:textStyle="bold"
        android:layout_marginLeft="10dp"
        android:paddingTop="30px"
        android:id="@+id/caloriesTxt"
        android:layout_gravity="left|top" />

```

```

<TextView
    style="@style/ButtonText"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="90px"

    android:textColor="@android:color/holo_blue_bright"
    android:textSize="22dp"
    android:textStyle="bold"
    android:layout_marginLeft="10dp"
    android:paddingTop="30px"
    android:id="@+id/fatTxt"
    android:layout_gravity="left|top" />

<TextView
    style="@style/ButtonText"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="120px"

    android:textColor="@android:color/holo_blue_bright"
    android:textSize="22dp"
    android:textStyle="bold"
    android:layout_marginLeft="10dp"
    android:paddingTop="50px"
    android:id="@+id/carbsTxt"
    android:layout_gravity="left|top" />

<TextView
    style="@style/ButtonText"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="160px"

    android:textColor="@android:color/holo_blue_bright"
    android:textSize="22dp"
    android:textStyle="bold"
    android:layout_marginLeft="10dp"
    android:paddingTop="60px"
    android:id="@+id/proteinTxt"
    android:layout_gravity="left|top" />

<ImageView
    style="@style/ButtonText"

```

```

        android:layout_width="75dp"
        android:layout_height="75dp"
        android:layout_marginBottom="50px"
        android:layout_marginLeft="60px"
        android:id="@+id/btn_identify"
        android:layout_gravity="left|bottom"
        android:src="@drawable/imgbtnpic"
        android:layout_above="@+id/textView"
        android:layout_alignParentStart="true"
        android:clickable="true" />

<ImageView
    style="@style/ButtonText"
    android:layout_width="75dp"
    android:layout_height="75dp"
    android:layout_marginBottom="50px"
    android:layout_marginRight="60px"
    android:id="@+id/btn_setting"
    android:layout_gravity="right|bottom"
    android:src="@drawable/setting"
    android:layout_above="@+id/textView"
    android:layout_alignParentStart="true"
    android:clickable="true" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="20px"
    android:layout_marginLeft="30px"
    android:textColor="@android:color/white"
    android:textStyle="bold"

    android:textAppearance="?android:attr/textAppearance
Small"
    android:text="IDENTIFY OBJECT"
    android:layout_gravity="left|bottom"
    android:id="@+id/identifyTxt"
    android:layout_alignParentBottom="true"
    android:layout_alignParentStart="true" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="20px"

```

```

        android:layout_marginRight="45px"
        android:textColor="@android:color/white"
        android:textStyle="bold"

        android:textAppearance="?android:attr/textAppearance
        Small"
        android:text="CONNECTION"
        android:layout_gravity="right|bottom"
        android:id="@+id/connectionTxt"
        android:layout_alignParentBottom="true"
        android:layout_alignParentStart="true" />

    <ImageButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/imgClose"
        android:layout_gravity="right|top"

        android:background="@android:drawable/ic_menu_close_
        clear_cancel"
        android:padding="20dp" />

</FrameLayout>

```

Kode Sumber 7.7. Kode Activity_main.xml

RANCANG BANGUN APLIKASI MEDIA INFORMASI NUTRISI PADA MAKANAN ATAU PRODUK MAKANAN MENGGUNAKAN TEKNOLOGI AUGMENTED REALITY BERBASIS ANDROID

Nama : Eko Adhi Wiyono
NRP : 5111100002
Jurusan : Teknik Informatika, FTIf, ITS
Dosen Pembimbing I : Wijayanti Nurul Khotimah, S.Kom.,
M.Sc.
Dosen Pembimbing II : Ridho Rahman Hariadi, S.Kom.,
M.Sc.

ABSTRAK

Obesitas mengalami kenaikan dua kali lipat semenjak tahun 1980 dan menjadi penyebab kematian nomor lima tertinggi pada setiap tahun. WHO (World Health Organization) mencatat bahwa sekitar 2.8 juta orang dewasa pada setiap tahunnya mengalami kematian yang disebabkan oleh obesitas. Hal tersebut dikarenakan manusia mengonsumsi makanan tanpa memperhatikan kebutuhan kalori dan kandungan nutrisi. Dengan mobilitas manusia yang semakin bertambah, sudah banyak dikembangkan aplikasi perangkat bergerak berbasis android yang mendukung untuk memantau kebutuhan kalori dan nutrisi makanan atau produk makanan seperti aplikasi Noom Coach, Calorie Counter, Lose It, dan lain sebagainya. Namun, aplikasi pada perangkat bergerak tersebut belum interaktif karena pengguna harus mencari terlebih dahulu nama makanan yang dikonsumsi kemudian dicari kandungan nutrisinya. Aplikasi pada Tugas Akhir ini menggunakan teknologi augmented reality teks berbasis android untuk menampilkan informasi nutrisi secara menarik dan informatif. Informasi yang ditampilkan berupa kalori, lemak, karbohidrat dan protein per takaran saji. Takaran saji

ditentukan oleh basis data Fatsecret. Melalui aplikasi ini, pengguna bisa mendapatkan informasi nutrisi hanya dengan mengambil gambar dari suatu objek makanan. Dengan menggunakan metode klasifikasi gambar, aplikasi bisa mengidentifikasi makanan atau produk makanan dari foto untuk mendapatkan informasi nama objek. Informasi nama objek tersebut dikirimkan ke webservice yang disediakan oleh FatSecret untuk mendapatkan informasi kandungan nutrisi. Berdasarkan hasil uji coba pada aplikasi Tugas Akhir ini, didapatkan rata-rata kebenaran dalam mengidentifikasi nama dan nutrisi sebesar 92%. Sedangkan rata-rata waktu yang diperlukan untuk identifikasi nama dan nutrisi yaitu 9.295 detik.

Kata kunci: *android, augmented reality, asupan nutrisi, informasi nutrisi, obesitas.*

DESIGN AND IMPLEMENTATION OF FOODS AND FOOD PRODUCTS NUTRITION USING AUGMENTED REALITY TECHNOLOGY IN ANDROID

Name	: Eko Adhi Wiyono
NRP	: 5111100002
Department	: Informatics Engineering, FTIf, ITS
Advisor I	: Wijayanti Nurul Khotimah, S.Kom., M.Sc.
Advisor II	: Ridho Rahman Hariadi, S.Kom., M.Sc.

ABSTRACT

Obesity has increased two-fold since 1980 and became the fifth highest cause of death in each year. WHO (World Health Organization) noted that about 2.8 million adults every year to experience deaths caused by obesity. That is because humans consume food without regard to the needs of calories and nutritional content. With the increasing mobility of people, has been developed android based mobile device applications that support to monitor caloric and nutritional needs food or food products such as application Coach Noom, Calorie Counter, Lose It, and etc. However, applications on mobile devices are not interactive because the user must seek first the name of the food consumed and then look for nutritional content. Applications in this final project using augmented reality technology android-based text to display nutritional information in an interesting and informative. The information displayed in the form of calories, fat, carbohydrate and protein per serving. Serving size is determined by FatSecret database. Through this application, users can get the nutritional information simply by taking a picture of an object food. By using the image classification method, an application can

identify the food or food products from photographs to obtain information object name. Information object name is sent to the web service provided by FatSecret to get nutritional information. Based on trial results on this final project application, obtained an average of truth in identifying the name and nutrition by 92%. While the average time required to identify the name and nutrients is 9.295 seconds.

Keywords: *android, augmented reality, nutrient intake, nutritional information, obesity.*

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Alhamdulillahirabbil'alamin, segala puji syukur kepada Allah SWT atas segala karunia dan rahmat-Nya penulis dapat menyelesaikan tugas akhir yang berjudul:

“Rancang Bangun Aplikasi Media Informasi Nutrisi pada Makanan atau Produk Makanan Menggunakan Teknologi Augmented Reality Berbasis Android”

Penulis ingin menyampaikan rasa hormat dan terima kasih kepada pihak-pihak yang telah membantu penulis dalam penyelesaian tugas akhir ini, terutama kepada:

1. Ayah Panggih Wiyono, ibunda Istianah, dan adik saya Ringga Arisgi Wiyono yang tiada hentinya setiap hari memberikan semangat secara lahir dan bathin. Adakalanya yang dicapai penulis hingga saat ini mungkin bukan karena dikabulkannya doa dari sang penulis, melainkan jawaban dari segala doa keluarga yang terus mengucur tiada henti untuk penulis.
2. Prof.Ir.Supeno Djanali, M.Sc., Ph.D. sebagai dosen wali penulis yang turut memberikan saran dan membimbing penulis dalam menentukan mata kuliah yang akan diambil pada setiap semester pembelajaran di Jurusan Teknik Informatika ITS selama ini.
3. Ibu Wijayanti Nurul Khotimah, S.Kom, M.Sc. dan Bapak Ridho Rahman Hariadi, S.Kom., M.Sc. yang telah bersedia untuk menjadi dosen pembimbing tugas akhir sehingga penulis dapat mengerjakan tugas akhir dengan arahan dan bimbingan yang baik hingga akhir.
4. Teman dan sekaligus keluarga kedua di kampus, Mahasiswa Teknik Informatika Angkatan 2011 yang telah berjuang bersama – sama selama menempuh pendidikan di

Jurusan ini. Setelah orang tua, kalian adalah alasan kedua saya bertahan di kampus tercinta ini.

5. Himpunan Mahasiswa Teknik Computer-Informatika ITS yang telah mempertemukan penulis dengan orang-orang hebat serta memberikan tempat untuk balas budi karena telah menjadikan Mahasiswa Teknik Informatika Angkatan 2011 sebagai keluarga.
6. Mas M. Misbachul Huda dan Ardhiansyah Baskara yang telah memberikan inspirasi dalam pencarian topik dan pengerjaan tugas akhir.
7. Ahmad Hayam Brilian yang telah mengajarkan tentang bagaimana pengolahan dan klasifikasi citra menggunakan bahasa Python.
8. Wilik dan Suliadi Marsetya yang telah mengajarkan tentang bagaimana implementasi client-server antara Android dan Webservice.
9. Muhammad Bagus Andra dan Putu Wiramaswara Widya yang telah mengajarkan tentang bagaimana menggunakan Flask Python sebagai Webservice.
10. Teman-teman di Lab. IGS, Rizaldi Tri Yanuar yang mau berbagi penggunaan komputer di lab, serta Rizka Wakhidatus Sholikhah, M. Chaqiqi Mudafi, Askary Muhammad, Didik Purwanto, M Iqbal Rustamadji sebagai teman seperjuangan pengerjaan tugas akhir di Lab. IGS.
11. Teman-teman satu bimbingan RMK IGS, Maranu Toto Negoro, Risal Andika Tridisaputra, Febry Amin Nur Hidayah, Mahendra Harsa Wardhana, Firda Ainur Ramadhani, Hashfi Alfian Ciyuda yang bersepakat untuk berjuang lulus bersama di satu anak bimbingan Ibu Wijayanti Nurul Khotimah, S.Kom., M.Sc.
12. Para sahabat huru hara selama kuliah, Astari Larasati, Farras Kinan, Monika Maytri, M. Aprialdi Rizky Pratama, Andina Triya Ramadhayanti, dan Friska Ayu Listya yang selalu menjadi tempat untuk berbagi sedih dan tawa di bangku perkuliahan.

13. Serta pihak–pihak lain yang turut membantu penulis baik secara langsung maupun tidak langsung, yang namanya tidak penulis sebutkan disini.

Penulis telah berusaha sebaik–baiknya dalam menyusun tugas akhir ini, mohon maaf apabila ada kesalahan dan kata-kata yang dapat menyinggung perasaan. Penulis berharap tugas akhir ini dapat menjadi media informasi nutrisi yang informatif dan dapat menjadi solusi dalam pengontrolan asupan gizi sehari–hari.

Surabaya, Juni 2015

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN.....	vii
ABSTRAK	ix
ABSTRACT	xi
KATA PENGANTAR.....	xiii
DAFTAR ISI	xvii
DAFTAR GAMBAR	xxi
DAFTAR TABEL	xxiii
DAFTAR KODE SUMBER	xxv
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Permasalahan.....	3
1.3. Batasan Permasalahan	4
1.4. Tujuan.....	5
1.5. Manfaat.....	5
1.6. Metodologi	5
1.6.1. Penyusunan Proposal Tugas Akhir.....	5
1.6.2. Studi Literatur.....	6
1.6.3. Analisis dan Desain Perangkat Lunak.....	6
1.6.4. Implementasi Perangkat Lunak	6
1.6.5. Pengujian dan Evaluasi	7
1.6.6. Penyusunan Buku Tugas Akhir	7
1.7. Sistematika Penulisan.....	7
BAB II TINJAUAN PUSTAKA	9
2.1. Nutrisi Makanan	9
2.2. Augmented Reality.....	16
2.3. Sistem Operasi Android dan Android Studio	18
2.4. SURF (<i>Speeded-Up Robust Features</i>).....	18
2.4.1. Pendeteksian <i>Keypoint</i>	19
2.4.2. Pendeskripsian <i>Keypoint</i>	20
2.4.3. Pencocokan Deskriptor.....	20
2.5. OpenCV.....	21
2.6. Platform Fatsecret API	22

2.7.	Webservice	23
2.8.	Flask Python	24
BAB III ANALISIS DAN PERANCANGAN SISTEM.....		25
3.1.	Analisis Perangkat Lunak	25
3.1.1.	Deskripsi Umum Perangkat Lunak	25
3.1.2.	Spesifikasi Kebutuhan Perangkat Lunak	26
3.1.3.	Identifikasi Pengguna	27
3.2.	Perancangan Perangkat Lunak	27
3.2.1.	Model Kasus Penggunaan	28
3.2.2.	Definisi Aktor	28
3.2.3.	Definisi Kasus Penggunaan	29
3.2.4.	Arsitektur Umum Sistem	33
3.2.5.	Rancangan Antarmuka Aplikasi	34
3.2.6.	Rancangan Proses Penggunaan Aplikasi	40
3.2.7.	Rancangan Proses Klasifikasi Gambar	42
BAB IV IMPLEMENTASI		45
4.1.	Lingkungan Pembangunan	45
4.1.1.	Lingkungan Pembangunan Perangkat Keras	45
4.1.2.	Lingkungan Pembangunan Perangkat Lunak	45
4.2.	Implementasi Antarmuka Aplikasi Android	46
4.2.1.	Implementasi Antarmuka Halaman Utama	46
4.2.2.	Implementasi Antarmuka Pengiriman Gambar ke <i>Webservice</i>	47
4.2.3.	Implementasi Antarmuka Rangkuman Nutrisi	48
4.2.4.	Implementasi Antarmuka <i>Server Connection</i>	49
4.2.5.	Implementasi Antarmuka Masukkan Alamat IP <i>Server</i>	50
4.3.	Implementasi Antarmuka <i>Webservice</i>	51
4.3.1.	Implementasi Lapisan <i>Server</i> pada <i>Webservice</i>	51
BAB V PENGUJIAN DAN EVALUASI		57
5.1.	Lingkungan Pembangunan	57
5.2.	Skenario Pengujian	57
5.2.1.	Pengujian Skenario 1 dan Evaluasi	59
5.2.2.	Pengujian Skenario 2 dan Evaluasi	61
5.2.3.	Pengujian Skenario 3 dan Evaluasi	63

5.2.4.	Pengujian Skenario 4 dan Evaluasi	65
5.2.5.	Pengujian Skenario 5 dan Evaluasi	68
5.2.6.	Pengujian Skenario 6 dan Evaluasi	72
5.2.7.	Pengujian Skenario 7 dan Evaluasi	75
5.2.8.	Pengujian Skenario 8 dan Evaluasi	77
5.2.9.	Pengujian Skenario 9 dan Evaluasi	81
5.2.10.	Pengujian Skenario 10 dan Evaluasi	83
5.2.11.	Pengujian Skenario 11 dan Evaluasi	86
5.2.12.	Pengujian Skenario 12 dan Evaluasi	88
5.2.13.	Pengujian Skenario 13 dan Evaluasi	91
5.2.14.	Pengujian Skenario 14 dan Evaluasi	93
BAB VI KESIMPULAN DAN SARAN.....		97
6.1.	Kesimpulan.....	97
6.2.	Saran.....	98
DAFTAR PUSTAKA		99
LAMPIRAN (A) KODE SUMBER.....		101
BIODATA PENULIS		115

DAFTAR TABEL

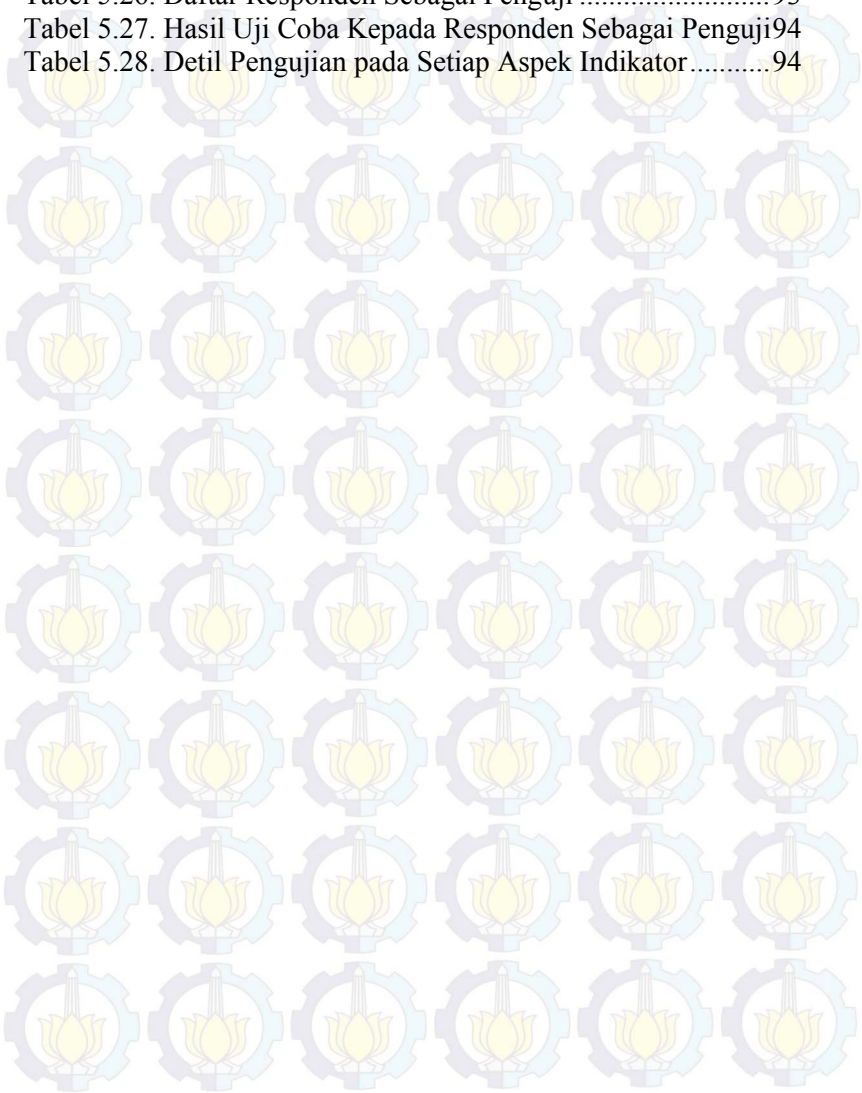
Tabel 2.1. Daftar Nutrisi Setiap Takaran Per 100gr.....	9
Tabel 3.1. Definisi Aktor.....	28
Tabel 3.2. Definisi Kasus Penggunaan.....	29
Tabel 3.3. Spesifikasi Kasus Penggunaan Mengambil Foto Objek Makanan atau Produk Makanan.....	30
Tabel 3.4. Spesifikasi Kasus Penggunaan Mendapatkan Informasi Nutrisi dalam Bentuk <i>Augmented Reality</i> Teks.....	31
Tabel 3.5. Spesifikasi Kasus Penggunaan Mengidentifikasi Foto Objek Makanan atau Produk Makanan.....	32
Tabel 5.1. Skenario Pengujian 1.....	59
Tabel 5.2. Skenario Pengujian 2.....	61
Tabel 5.3. Skenario Pengujian 3.....	63
Tabel 5.4. Dokumentasi Hasil Pengujian 3.....	64
Tabel 5.5. Skenario Pengujian 4.....	66
Tabel 5.6. Dokumentasi Hasil Pengujian 4.....	67
Tabel 5.7. Skenario Pengujian 5.....	69
Tabel 5.8. Dokumentasi Hasil Pengujian 5.....	70
Tabel 5.9. Skenario Pengujian 6.....	72
Tabel 5.10. Dokumentasi Hasil Pengujian 6.....	73
Tabel 5.11. Skenario Pengujian 7.....	75
Tabel 5.12. Dokumentasi Hasil Pengujian 7.....	76
Tabel 5.13. Skenario Pengujian 8.....	77
Tabel 5.14. Dokumentasi Hasil Pengujian 8.....	78
Tabel 5.15. Skenario Pengujian 9.....	81
Tabel 5.16. Dokumentasi Hasil Pengujian 9.....	82
Tabel 5.17. Skenario Pengujian 10.....	83
Tabel 5.18. Dokumentasi Hasil Pengujian 10.....	84
Tabel 5.19. Skenario Pengujian 11.....	86
Tabel 5.20. Dokumentasi Hasil Pengujian 11.....	87
Tabel 5.21. Skenario Pengujian 12.....	88
Tabel 5.22. Dokumentasi Hasil Pengujian 12.....	89
Tabel 5.23. Hasil Pengujian 13.....	91
Tabel 5.24. Hasil Kalkulasi Prosentase Kebenaran.....	92

Tabel 5.25. Hasil Kalkulasi Waktu Identifikasi92

Tabel 5.26. Daftar Responden Sebagai Penguji 93

Tabel 5.27. Hasil Uji Coba Kepada Responden Sebagai Penguji94

Tabel 5.28. Detil Pengujian pada Setiap Aspek Indikator 94



DAFTAR GAMBAR

Gambar 2.1. <i>Augmented Reality</i> dengan Objek Buah Apel	17
Gambar 2.2. Pendeteksian <i>Keypoint Point</i>	19
Gambar 2.3. Pendeskripsian <i>Keypoint</i>	20
Gambar 2.4. Visualisasi Pencocokan Deskriptor Gambar	21
Gambar 2.5. Contoh Penggunaan <i>Fatsecret</i> untuk Pencarian Nutrisi Buah Apel.....	23
Gambar 3.1. Model Kasus Penggunaan Aplikasi	28
Gambar 3.2. Desain Arsitektur Aplikasi	34
Gambar 3.3. Rancangan Antarmuka Halaman Utama	35
Gambar 3.4. Rancangan Antarmuka Pengiriman Gambar ke Server	36
Gambar 3.5. Rancangan Antarmuka Rangkuman Nutrisi	37
Gambar 3.6. Rancangan Antarmuka <i>Server Connection</i>	38
Gambar 3.7. Rancangan Antarmuka Masukkan Alamat IP <i>Server</i>	39
Gambar 3.8. Rancangan Antarmuka <i>Webservice</i>	40
Gambar 3.9. Diagram Alir Proses Penggunaan Aplikasi	41
Gambar 3.10. Diagram Alir Proses Klasifikasi Gambar	42
Gambar 3.11. Proses Ekstraksi Fitur <i>Keypoint</i> Gambar	43
Gambar 3.12. Proses Perhitungan Kemiripan Gambar	44
Gambar 4.1. Antarmuka Halaman Utama	47
Gambar 4.2. Antarmuka Pengiriman Gambar ke <i>Webservice</i>	48
Gambar 4.3. Antarmuka Rangkuman Nutrisi.....	49
Gambar 4.4. Antarmuka <i>Server Connection</i>	50
Gambar 4.5. Antarmuka Masukkan Alamat IP <i>Server</i>	51
Gambar 5.1. Modus Pengambilan Gambar Objek Makanan atau Produk Makanan	60
Gambar 5.2. Pengujian Tombol <i>Identify Object</i>	62
Gambar 5.3. Pengujian Tombol <i>Connection</i>	62
Gambar 5.4. Contoh Uji Coba Pengujian Aplikasi pada Buah Pisang	65
Gambar 5.5. Identifikasi Kemiripan Buah Jeruk dan Pisang	67

Gambar 5.6. Contoh Uji Coba Pengujian Aplikasi pada Buah Jeruk	68
Gambar 5.7. Identifikasi Kemiripan Buah Apel dan Jeruk	70
Gambar 5.8. Identifikasi Kemiripan Buah Apel dan Pisang	71
Gambar 5.9. Contoh Uji Coba Pengujian Aplikasi pada Buah Apel	72
Gambar 5.10. Contoh Uji Coba Pengujian Aplikasi pada Produk Indomie.....	74
Gambar 5.11. Contoh Uji Coba Pengujian Aplikasi pada Produk Pringles.....	77
Gambar 5.12. Contoh Uji Coba Pengujian Aplikasi pada Buah Lemon.....	79
Gambar 5.13. Identifikasi Kemiripan Buah Lemon dan Pisang ..	80
Gambar 5.14. Identifikasi Kemiripan Buah Lemon dan Apel.....	80
Gambar 5.15. Contoh Uji Coba Pengujian Aplikasi pada Produk Pocky	83
Gambar 5.16. Contoh Uji Coba Pengujian Aplikasi pada Produk Oreo	85
Gambar 5.17. Contoh Uji Coba Pengujian Aplikasi pada Produk Snickers	88
Gambar 5.18. Contoh Uji Coba Pengujian Aplikasi pada Produk Tango.....	90

DAFTAR KODE SUMBER

Kode Sumber 2.1. Contoh Kode Program untuk Mengakses Basis Data Nutrisi <i>Fatsecret</i>	22
Kode Sumber 2.2. Contoh Penggunaan Flask Python.....	24
Kode Sumber 4.1. Implementasi <i>Server</i> pada <i>Webservice</i>	52
Kode Sumber 4.2. Implementasi Ekstraksi Fitur <i>Keypoint</i>	53
Kode Sumber 4.3. Implementasi Penyimpanan Hasil Ekstraksi Fitur	54
Kode Sumber 4.4. Implementasi Pembacaan Hasil Ekstraksi Fitur Gambar dari Android Untuk Dikomputasi	55
Kode Sumber 4.5. Implementasi Pencocokan Gambar di Basis data Direktori	56
Kode Sumber 4.6. Implementasi Mendapatkan Rangkuman Nutrisi Objek Makanan	56
Kode Sumber 7.1. Kode <i>MainActivity.java</i> (Bagian 1).....	102
Kode Sumber 7.2. Kode <i>MainActivity.java</i> (Bagian 2).....	103
Kode Sumber 7.3. Kode <i>MainActivity.java</i> (Bagian 3).....	104
Kode Sumber 7.4. Kode <i>MainActivity.java</i> (Bagian 4).....	107
Kode Sumber 7.5. Kode <i>MainActivity.java</i> (Bagian 5).....	108
Kode Sumber 7.6. Kode <i>CameraView.java</i>	109
Kode Sumber 7.7. Kode <i>Activity_main.xml</i>	113

BIODATA PENULIS



Penulis lahir di Jember, 29 April 1992. Penulis menempuh pendidikan di SD Negeri 3 Menampu (1999-2005), SMP Negeri 1 Puger (2005-2008), dan SMA Negeri 1 Jember (2008-2011). Penulis melanjutkan pendidikan sarjana di Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember Surabaya. Dalam menyelesaikan pendidikan S1, penulis mengambil bidang minat Interaksi Grafika dan Seni (IGS). Selain aktif sebagai mahasiswa di bidang akademik, penulis juga aktif dalam berbagai kegiatan nonakademik. Tahun ke-2 sebagai mahasiswa, penulis aktif sebagai Staff Departemen Pengembangan Sumber Daya Mahasiswa dan sebagai Organizer Committee kaderisasi Kabinet HMTC Bersatu. Penulis aktif dalam kepanitiaan National Seminar Technology - SCHEMATICS 2012 ITS dan juga aktif di bidang sosial sebagai Volunteer TC Mengabdi di Desa Tempurejo, Surabaya. Tahun ketiga sebagai mahasiswa, penulis aktif sebagai Staff Ahli Departemen Sumber Daya Mahasiswa serta sebagai Instructor Committee kaderisasi Kabinet HMTC Bersahabat dan GERIGI ITS 2013. Penulis juga aktif sebagai panitia National Seminar Technology - SCHEMATICS 2013 ITS. Penulis memiliki pengalaman kerja praktik selama satu bulan di Biro Sistem Informasi PT. Wijaya Karya (Persero) Tbk, Jakarta Timur sebagai analis sistem untuk penambahan kebutuhan aplikasi WIKA ScoreCard. Penulis dapat dihubungi melalui alamat email ekoadhiwiyono@gmail.com

BAB I

PENDAHULUAN

Pada bab ini dipaparkan mengenai garis besar tugas akhir yang meliputi latar belakang, tujuan, rumusan dan batasan permasalahan, metodologi pembuatan tugas akhir, dan sistematika penulisan.

1.1. Latar Belakang

Obesitas sudah mengalami kenaikan dua kali lipat semenjak tahun 1980 dan menjadi penyebab kematian nomor lima tertinggi pada setiap tahun. Sekitar 2.8 juta orang dewasa pada setiap tahunnya mengalami kematian yang disebabkan oleh obesitas. Badan Organisasi Kesehatan Dunia WHO (*World Health Organization*) mendefinisikan bahwa seseorang mengalami kelebihan berat badan apabila indeks massa berat lebih dari 25 kg/m² dan menjadi obesitas apabila indeks massa berat melebihi 30 kg/m². Dalam rangka memerangi obesitas, WHO (*World Health Organization*) merekomendasikan bahwa setiap individu harus mengurangi kalori dari lemak dan gula berlebih, serta menjaga pola hidup sehat dengan berolahraga secara teratur minimal 150 menit per minggu [1]. Oleh karena itu, setiap individu disarankan untuk memperhatikan kandungan nutrisi pada makanan yang dikonsumsi sehingga bisa mengurangi kalori dari lemak dan gula darah berlebih.

Nutrisi adalah bidang ilmu yang mempelajari bagaimana makanan dapat mempengaruhi kesehatan dan kelangsungan hidup manusia. Manusia memerlukan makanan untuk tumbuh, bereproduksi, dan memelihara kesehatan yang baik. Tanpa makanan, tubuh manusia tidak dapat menjaga suhunya, membangun atau memperbaiki jaringan, atau memelihara detak jantung. Memakan makanan yang benar dapat menghindarkan manusia dari berbagai penyakit atau sembuh lebih cepat ketika

penyakit menyerang. Faktor dan fungsi penting lain dipenuhi dengan zat-zat kimia di dalam makanan manusia yang disebut nutrisi. Nutrisi dikategorikan sebagai karbohidrat, protein, lemak, vitamin-vitamin dan mineral [2]. Berat badan dapat diturunkan dengan menggunakan cara membatasi asupan nutrisi. Faktor pengali untuk energi yang umum diterima oleh banyak orang adalah sebagai berikut: 1 gram karbohidrat menghasilkan 4 kkal, 1 gram protein 4 kkal, dan 1 gram lemak kkal, sedangkan alkohol 7 kkal [3].

Kalori merupakan satuan tenaga yang akan dibakar oleh tubuh. Banyaknya jumlah kalori yang dibutuhkan dapat berbeda-beda pada setiap orang. Kebutuhan akan energi yang berbeda dipengaruhi oleh berat badan, jenis kelamin, umur dan aktivitas fisik yang dilakukan. Menentukan kebutuhan kalori dan asupan nutrisi tubuh per hari merupakan suatu kendala bagi kita. Kendala yang sering dihadapi adalah banyak sekali produk-produk makanan yang belum diketahui total kalori dan nutrisi yang terkandung di dalamnya. Dengan mobilitas manusia yang semakin bertambah, sudah banyak dikembangkan perangkat bergerak berbasis android yang mendukung untuk memantau kebutuhan kalori dan nutrisi makanan atau produk makanan. Misalnya saja aplikasi *Nutrition Facts* [4], aplikasi ini memberikan informasi berupa basis data nutrisi dan kalori yang tersimpan dalam suatu makanan atau produk makanan. Pengguna aplikasi ini dihadapkan pada fitur *search* untuk mencari kandungan nutrisi berdasarkan makanan atau produk makanan yang dipilih. Informasi nutrisi makanan yang dipilih berupa kandungan air, kalori, protein, total lemak, dan karbohidrat. Namun kekurangan dari aplikasi ini adalah pengguna tidak bisa secara langsung mendapatkan informasi kandungan nutrisi. Pengguna harus mencari terlebih dahulu makanan atau produk makanan apa yang akan dikonsumsi. Pengguna juga terbatas pada basis data makanan atau produk makanan aplikasi tersebut saja.

Perkembangan *platform android* sudah menggunakan teknologi *augmented reality* untuk mendukung kinerja aplikasi.

Perkembangan *augmented reality* semakin hari semakin pesat. Dari aplikasi yang sederhana hingga aplikasi yang kompleks. Misalnya saja aplikasi *augmented reality* untuk pemasaran produk, rehabilitasi pasca sakit *stroke*, hingga *game 3D* yang dapat berinteraksi dengan pengguna. Teknologi *augmented reality* dirasa masih baru dan menarik bagi masyarakat. Beberapa riset menemukan bahwa implementasi dari *augmented reality* pada aplikasi perangkat bergerak bisa merepresentasikan informasi secara visual [5]. Oleh karena itu, maka pada Tugas Akhir ini diimplementasikan *augmented reality* untuk merepresentasikan informasi nutrisi secara menarik.

Rancang bangun media informasi nutrisi pada makanan atau produk makanan pada tugas akhir ini, menggunakan teknologi *augmented reality* untuk menampilkan informasi nutrisi. Informasi yang ditampilkan berupa kandungan nutrisi yaitu kalori, lemak, karbohidrat dan protein. Misalnya pengguna aplikasi ini akan mengonsumsi 1 buah apel, 2 lembar roti tawar, dan satu gelas susu. Untuk mengetahui kandungan nutrisi di dalam makanan atau produk makanan tersebut, pengguna bisa mendapatkan informasi dengan mengambil foto makanan atau produk makanan. Aplikasi akan menampilkan informasi nutrisi berupa kalori, lemak, karbohidrat dan protein objek yang difoto dalam bentuk *augmented reality*.

Aplikasi ini dibangun dengan harapan pengguna perangkat bergerak bisa mengontrol makanan atau produk makanan apa saja yang akan dimakan dengan memperhatikan kandungan nutrisi di dalamnya kapanpun dan dimanapun. Dengan teknologi *augmented reality* yang masih baru di khalayak umum diharapkan bisa menarik perhatian pengguna untuk memperhatikan lebih kalori, karbohidrat, protein dan lemak pada makanan atau produk makanan yang akan dikonsumsi sesuai dengan kebutuhan nutrisi di setiap harinya.

1.2. Rumusan Permasalahan

Rumusan masalah yang diangkat dalam tugas akhir ini adalah sebagai berikut:

1. Bagaimana mengidentifikasi jenis makanan atau produk makanan dari masukan berupa gambar serta mendapatkan informasi nutrisi berupa kalori, karbohidrat, protein dan lemak yang terkandung?
2. Bagaimana menampilkan informasi nutrisi berupa kalori, karbohidrat, protein dan lemak terkandung di dalam makanan kepada pengguna perangkat bergerak berbasis Android yang menarik menggunakan teknologi *augmented reality* teks?

1.3. Batasan Permasalahan

Permasalahan yang dibahas dalam tugas akhir ini memiliki beberapa batasan, di antaranya sebagai berikut:

1. Aplikasi dibangun khusus untuk perangkat bergerak dengan sistem operasi Android dengan versi minimum 4.0 yang terhubung dengan internet.
2. Aplikasi hanya bisa mengidentifikasi nutrisi suatu makanan atau produk makanan yang sudah dilatih sebelumnya.
3. Aplikasi menggunakan informasi nutrisi yang terkandung dalam makanan atau produk makanan menggunakan basis data *FatSecret* (www.fatsecret.com).
4. Aplikasi hanya dapat dijalankan pada jaringan lokal yang terhubung antara aplikasi dan *webservice*.
5. Aplikasi hanya dapat berkomunikasi dengan *webservice* apabila *Windows Firewall* pada *server* dalam kondisi tidak aktif.
6. Jarak antara kamera dan objek makanan dalam pengambilan gambar maksimal 15-20cm, serta tidak memiliki latar belakang objek yang terlalu ramai.
7. *Augmented relaity* teks berupa kandungan nutrisi yang hanya bisa muncul di layar modus kamera.

1.4. Tujuan

Tujuan dalam pembuatan tugas akhir ini antara lain adalah sebagai berikut:

1. Membuat aplikasi yang bisa mengidentifikasi jenis makanan atau produk makanan dari masukan berupa gambar serta mendapatkan informasi nutrisi berupa kalori, karbohidrat, protein dan lemak yang terkandung.
2. Menampilkan informasi nutrisi berupa kalori, karbohidrat, protein dan lemak terkandung di dalam makanan atau produk makanan kepada pengguna perangkat bergerak berbasis Android menggunakan teknologi *augmented reality* teks sehingga lebih menarik.

1.5. Manfaat

Dengan dibangunnya perangkat lunak media informasi nutrisi pada makanan atau produk makanan menggunakan teknologi *augmented reality* teks berbasis Android diharapkan kita bisa lebih memerhatikan kandungan nutrisi makanan yang akan dikonsumsi terutama kalori, karbohidrat, lemak dan protein sesuai dengan kebutuhan setiap hari. Dikemas dengan teknologi *augmented reality* teks diharapkan bisa lebih menarik pengguna aplikasi serta bisa menjadi media informasi nutrisi yang informatif.

1.6. Metodologi

Tahapan-tahapan yang dilakukan dalam pengerjaan tugas akhir ini adalah sebagai berikut.

1.6.1. Penyusunan Proposal Tugas Akhir.

Tahap awal pengerjaan tugas akhir adalah penyusunan proposal. Proposal ini mengajukan gagasan mengenai “Rancang Bangun Aplikasi Media Informasi Nutrisi pada Makanan atau Produk Makanan Menggunakan Teknologi *Augmented Reality* Berbasis Android”. Pada tahap ini dilakukan pengumpulan dan penggalan informasi dan studi literatur yang diperlukan dalam proses perancangan dan implementasi sistem yang akan dibangun.

1.6.2. Studi Literatur

Studi literatur yang dilakukan dalam pengerjaan tugas akhir ini adalah dengan mengumpulkan informasi yang diperlukan yaitu berupa:

- i. Penggunaan metode klasifikasi gambar untuk mengenali objek dengan mengambil foto objek;
- ii. Pengembangan *webservice* untuk mengidentifikasi foto objek dan mengirimkan informasi objek berupa nama objek;
- iii. Penggunaan *Fatsecret* API sebagai basis data penyimpanan informasi nutrisi makanan atau produk makanan;
- iv. Penerapan *augmented reality* teks untuk menampilkan informasi nutrisi berupa kalori, karbohidrat, protein dan lemak;
- v. Studi literatur dapat didapatkan melalui buku tutorial, artikel ilmiah, maupun *website*.

1.6.3. Analisis dan Desain Perangkat Lunak

Analisis dan desain perangkat lunak dilakukan bergantung dari hasil analisa kebutuhan sistem. Aplikasi media informasi nutrisi pada makanan atau produk makanan menggunakan teknologi *augmented reality* berbasis android. Aplikasi mampu mengidentifikasi objek untuk mendapatkan informasi nutrisi dari objek yang difoto. Kemudian informasi nutrisi ditampilkan ke layar perangkat bergerak pengguna menggunakan *augmented reality* teks.

1.6.4. Implementasi Perangkat Lunak

Pada tahap implementasi dilakukan pembuatan perangkat lunak yang merupakan implementasi dari rancangan yang telah dibuat. Pada tahap ini dilakukan implementasi klasifikasi gambar untuk mengenali objek dengan mengambil foto objek. Dilanjutkan dengan implementasi *augmented reality* teks untuk menampilkan informasi nutrisi berupa kalori, karbohidrat, lemak dan protein.

1.6.5. Pengujian dan Evaluasi

Tahap ini dilakukan pengujian terhadap sistem menggunakan data atau kasus yang telah disiapkan. Uji coba dilakukan untuk menguji fungsionalitas sistem, mencari masalah yang mungkin timbul pada sistem dan melakukan perbaikan apabila terdapat kekurangan. Uji coba ini dilakukan ke mahasiswa dan masyarakat sekitar ITS kecuali dosen dan staf tata usaha yang aktif sebagai konsumen makanan dan produk makanan saja. Uji coba yang dilakukan melibatkan remaja umur 13 – 21 tahun dan dewasa umur 21 – 40 tahun yang memiliki perangkat bergerak berbasis *android*.

1.6.6. Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat.

1.7. Sistematika Penulisan

Buku tugas akhir ini disusun dengan sistematika penulisan sebagai berikut.

BAB I. PENDAHULUAN

Bab ini berisi mengenai latar belakang, tujuan, dan manfaat dari pembuatan tugas akhir. Selain itu, permasalahan, batasan masalah, metodologi yang digunakan, dan sistematika penulisan juga merupakan bagian dari bab ini.

BAB II. TINJAUAN PUSTAKA

Bab ini berisi penjelasan secara detail mengenai dasar-dasar penunjang dan teori-teori yang digunakan untuk mendukung pembuatan tugas akhir ini.

BAB III. ANALISIS DAN PERANCANGAN SISTEM

Bab ini membahas tahap analisis permasalahan dan perancangan dari sistem yang akan dibangun. Analisis permasalahan membahas permasalahan yang diangkat dalam pengerjaan tugas akhir.

BAB IV. IMPLEMENTASI

Bab ini membahas implementasi dari desain yang telah dibuat pada bab sebelumnya. Bab ini berisi proses implementasi hasil analisis pada semua modul.

BAB V. PENGUJIAN DAN EVALUASI

Bab ini menjelaskan kemampuan perangkat lunak dengan melakukan pengujian kebenaran dan pengujian kinerja dari sistem yang telah dibuat.

BAB VI. KESIMPULAN DAN SARAN

Bab ini merupakan bab terakhir yang menyampaikan kesimpulan dari hasil uji coba yang dilakukan dan saran untuk pengembangan perangkat lunak ke depannya.

BAB II

TINJAUAN PUSTAKA

Pada bab ini berisi penjelasan teori-teori yang berkaitan dengan metode yang diajukan pada pengimplementasian perangkat lunak. Penjelasan ini bertujuan untuk memberikan gambaran secara umum terhadap sistem yang dibuat dan berguna sebagai penunjang dalam pengembangan perangkat lunak.

2.1. Nutrisi Makanan

Nutrisi adalah zat dalam makanan yang dibutuhkan organisme untuk dapat tumbuh dan berkembang dengan baik sesuai dengan fungsinya. Nutrisi dalam jumlah yang memadai dan sesuai dengan kebutuhan akan memberikan energi bagi tubuh untuk dapat tumbuh dan berkembang serta memperbaiki jaringan yang rusak. Kekurangan nutrisi akan membuat tubuh organisme tidak tumbuh dan berkembang sesuai dengan takdirnya, bahkan dapat menyebabkan penyakit hingga berakhir dengan kematian. Terganggunya proses metabolisme tubuh merupakan gejala awal kekurangan nutrisi [6]. Nutrisi berimbang terdiri dari 30% lemak 30% protein 40% karbohidrat. Fungsi Nutrisi adalah sebagai sumber energi, pendukung dan pengatur proses metabolisme, menjaga keseimbangan metabolisme, pembentuk sel-sel jaringan tubuh, memperbaiki sel-sel yang rusak, mempertahankan fungsi organ tubuh, dan lain sebagainya [6]. Daftar nutrisi setiap takaran per 100gr dapat dilihat pada Tabel 2.1.

Tabel 2.1. Daftar Nutrisi Setiap Takaran Per 100gr

No.	Makanan	Protein	Karbohidrat	Lemak	Kalori
1	Beras Merah	3 gr	23 gr	1 gr	111
2	Beras Putih	2 gr	21 gr	0	97

No.	Makanan	Protein	Karbohidrat	Lemak	Kalori
3	Kentang rebus	2 gr	20 gr	0	86
4	Kentang goreng (French Fries)	4 gr	38 gr	17 gr	319
5	Ubi rebus (sweet potato)	1 gr	18 gr	0	76
6	Ubi panggang	2 gr	21 gr	0	90
7	Jagung rebus	3 gr	25 gr	1 gr	108
8	Snack Jagung (Corn Flake)	2 gr	24 gr	0	101
9	Popcorn bumbu karamel	4 gr	79 gr	13 gr	431
10	Mi Instant	10 gr	53 gr	17 gr	400
11	Spaghetti	6 gr	31 gr	1 gr	157
12	Alpukat	2 gr	9 gr	15 gr	160
13	Anggur	1 gr	8 gr	0	32
14	Apel dengan kulit	0	14 gr	0	52
15	Apel tanpa kulit	0	13 gr	0	48
16	Blewah	1 gr	8 gr	0	34
17	Blueberry	1 gr	12 gr	0	57
18	Buah Naga	1 gr	14 gr	1 gr	50
19	Durian	1gr	27 gr	5 gr	147
20	Jambu biji	3 gr	14 gr	1 gr	68
21	Jeruk	1 gr	12 gr	0	47
22	Jus Jeruk	1 gr	10 gr	0	45
23	Kelapa (daging kelapa)	3 gr	15 gr	33 gr	354
24	Kelapa (air kelapa)	1 gr	4 gr	0	19
25	Kelapa (santan mentah)	2 gr	6 gr	24 gr	230
26	Kesemek	1 gr	19 gr	0	70
27	Kismis	3 gr	79 gr	0	299

No.	Makanan	Protein	Karbohidrat	Lemak	Kalori
28	Kiwi	1 gr	15 gr	1 gr	61
29	Leci	1 gr	17 gr	0	66
30	Lemon	1 gr	9 gr	0	29
31	Mangga	1 gr	17 gr	0	65
32	Markisa	2 gr	23 gr	1 gr	97
33	Melon	1 gr	9 gr	0	36
34	Mentimun (dengan kulit)	1 gr	4 gr	0	15
35	Mentimun acar	0	2 gr	0	11
36	Nanas	1 gr	13 gr	0	50
37	Nangka	1 gr	24 gr	0	94
38	Pear	0	11 gr	0	42
39	Pepaya	1 gr	10 gr	0	39
40	Persik (peach)	1 gr	10 gr	0	39
41	Pisang	1 gr	23 gr	0	89
42	Keripik Pisang	2 gr	58 gr	34 gr	519
43	Sawo	0	20 gr	1 gr	83
44	Semangka	1 gr	8 gr	0	30
45	Strawberi	1 gr	8 gr	0	32
46	Srikaya	1 gr	17 gr	0	66
47	Asparagus matang tanpa garam	2 gr	4 gr	0	22
48	Banguang	1 gr	9 gr	0	38
49	Brokoli rebus	1 gr	4 gr	1 gr	22
50	Buncis rebus	2 gr	8 gr	0	35
51	Bunga kol	2 gr	4 gr	0	23
52	Cabai hijau pedas mentah	2 gr	9 gr	0	40
53	Cabai merah pedas mentah	2 gr	9 gr	0	40

No.	Makanan	Protein	Karbohidrat	Lemak	Kalori
54	Jamur kuping kering	9 gr	73 gr	1 gr	284
55	Kedele rebus	17 gr	10 gr	1 gr	173
56	Kubis mentah	1 gr	6 gr	0	25
57	Labu matang tanpa garam	1 gr	5 gr	0	20
58	Pare rebus	1 gr	4 gr	0	19
59	Peterseli (parsley)	3 gr	6 gr	1 gr	36
60	Tauge rebus	2 gr	4 gr	0	21
61	Rebung rebus dengan garam	2 gr	2 gr	0	11
62	Rebung rebus tanpa garam	2 gr	2 gr	0	12
63	Sayur asin kalengan	1 gr	5 gr	0	19
64	Seledri matang tanpa garam	1 gr	4 gr	0	18
65	Selada (home prepared)	1 gr	12 gr	3 gr	78
66	Terong matang tanpa garam	1 gr	9 gr	0	35
67	Terong matang dengan garam	1 gr	8 gr	0	33
68	Tomat Merah mentah	1 gr	4 gr	0	18
69	Tomat Kuning mentah	1 gr	3 gr	0	15
70	Wortel mentah	1 gr	10 gr	0	41
71	Jus wortel	1 gr	9 gr	0	40
72	Cod Atlantik	23 gr	0	1 gr	105
73	Cod Pasifik	23 gr	0	1 gr	105
74	Cumi-cumi (goreng)	18 gr	8 gr	7 gr	175
75	Gurame	23 gr	0	7 gr	162
76	Gurita	30 gr	4 gr	2 gr	164
77	Herring Atlantik	33 gr	0	17 gr	290

No.	Makanan	Protein	Karbohidrat	Lemak	Kalori
78	Herring Pasifik	21 gr	0	18 gr	250
79	Ikan asin mackerel	18 gr	0	25 gr	305
80	Kakap	26 gr	0	2 gr	128
81	Keong ukuran sedang (jenis whelk)	48 gr	16 gr	1 gr	275
82	Keong besar jenis conch (panggang)	26 gr	2 gr	1 gr	130
83	Kepah	24 gr	7 gr	4 gr	172
84	Kepiting (biru)	20 gr	0	2 gr	102
85	Kijing (remis besar)	26 gr	5 gr	2 gr	148
86	Lele	19 gr	0	8 gr	152
87	Lobster	26 gr	3 gr	2 gr	143
88	Mackerel Atlantik	24 gr	0	18 gr	262
89	Mackerel Pasifik	26 gr	0	10 gr	201
90	Remis	23 gr	0	1 gr	112
91	Salmon	25 gr	0	8 gr	182
92	Sarden Atlantik kaleng	25 gr	0	11 gr	208
93	Sarden Pasifik kaleng saus tomat	21 gr	1 gr	10 gr	186
94	Sotong	32 gr	2 gr	1 gr	158
95	Teri goreng (anchovy)	29 gr	0	10 gr	210
96	Tiram jenis abalone (goreng)	20 gr	11 gr	7 gr	189
97	Tiram jenis eastern oyster	14 mg	8 gr	5 gr	137
98	Tiram jenis Pasifik oyster	19 gr	10 gr	5 gr	163
99	Trout (ikan air tawar)	27 gr	0	8 gr	190
100	Tuna	30 gr	0	6 gr	184
101	Udang (shrimp)	21 gr	0	1 gr	99

No.	Makanan	Protein	Karbohidrat	Lemak	Kalori
102	Telur rebus dengan cangkang (hard-boiled)	13 gr	1 gr	11 gr	155
103	Telur rebus tanpa cangkang/ dipecahkan (poached)	13 gr	1 gr	10 gr	142
104	Omelet	11 gr	1 gr	12 gr	157
105	Telur ceplok	14 gr	1 gr	15 gr	196
106	Telur orak arik	11 gr	2 gr	12 gr	167
107	Eggnog	4 gr	14 gr	7 gr	135
108	Putih telur mentah segar	11 gr	1 gr	0	48
109	Kuning telur mentah segar	16 gr	4 gr	27 gr	317
110	Telur mentah segar utuh	13 gr	1 gr	10 gr	143
111	Telur bebek mentah segar utuh	13 gr	1 gr	14 gr	185
112	Telur angsa mentah segar utuh	14 gr	1 gr	13 gr	185
113	Telur puyuh mentah segar utuh	13 gr	0	11 gr	158
114	Telur kalkun mentah segar utuh	14 gr	1 gr	12 gr	171
115	Roti gandum utuh (whole-wheat bread)	13 gr	41 gr	3 gr	247
116	Roti gandum utuh panggang (whole-wheat bread, toasted)	16 gr	41 gr	4 gr	306
117	Roti putih panggang (white bread, toasted)	9 gr	54 gr	4 gr	293
118	Roti bagel telur	11 gr	53 gr	2 gr	278
119	Roti bagel gandum	11 gr	53 gr	1 gr	255
120	Tempe Rebus	18 gr	9 gr	11 gr	196

No.	Makanan	Protein	Karbohidrat	Lemak	Kalori
121	Tofu goreng	17 gr	10 gr	20 gr	271
122	Daging sapi asap (sliced)	19 gr	1 gr	6 gr	139
123	Daging kambing panggang	27 gr	0	3 gr	143
124	Daging kerbau panggang	27 gr	0	2 gr	131
125	Sosis sapi matang	18 gr	0	28 gr	332
126	Corned sapi (kaleng)	19 gr	1 gr	7 gr	142
127	Donat dengan coklat atau gula	5 gr	57 gr	20 gr	417
128	Kue bolu	5 gr	61 gr	3 gr	289
129	Brownies	5 gr	64 gr	16 gr	405
130	Muffin panggang dengan kismis	5 gr	56 gr	9 gr	313
131	Croissant keju	9 gr	47 gr	21 gr	414
132	Waffle panggang	7 gr	49 gr	10 gr	312
133	Biskuit plain komersial	6 gr	48 gr	17 gr	365
134	Coklat wafer	7 gr	72 gr	14 gr	433
135	Cracker meal	9 gr	81	2 gr	383
136	Pie buah (kering)	3 gr	43 gr	16 gr	316
137	Pie apel	2 gr	34 gr	11 gr	237
138	Puding coklat manis	2 gr	23 gr	5 gr	142
139	Kue terang bulan plain	10 gr	71 gr	5 gr	376
140	Kue pastry buah	5 gr	69 gr	11 gr	391
141	Keripik pisang	2 gr	58 gr	34 gr	519
142	Keripik kentang tanpa garam	7 gr	53 gr	35 gr	536
143	Keripik kentang dengan garam	7 gr	50 gr	37 gr	547

No.	Makanan	Protein	Karbohidrat	Lemak	Kalori
144	Cemilan kedelai dengan garam	27 gr	53 gr	7 gr	376
145	Susu bubuk	26 gr	38 gr	27 gr	496
146	Susu kental manis	8 gr	54 gr	9 gr	321
147	Keju mozarella (whole milk)	22 gr	2 gr	22 gr	300
148	Keju Parmesan	38 gr	4 gr	29 gr	431
149	Keju cheddar	25 gr	1 gr	33 gr	403
150	Krim bubuk	5 gr	55	35 gr	545

Sumber: Tabel Nutrisi Dunia Fitnes, 2014

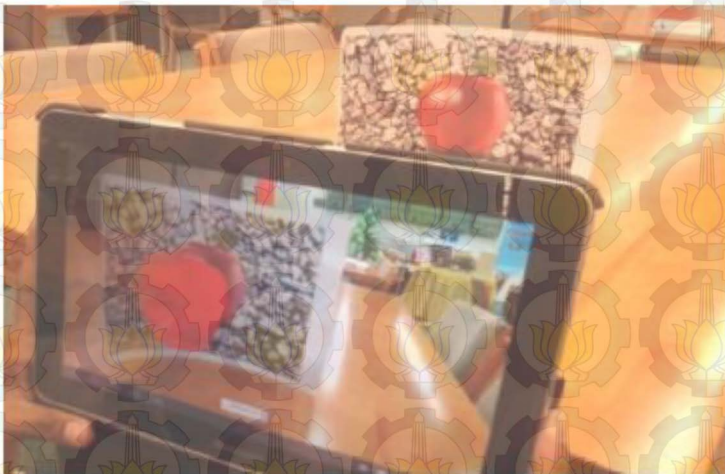
Kalori sesungguhnya sangat dibutuhkan tubuh untuk menghasilkan energi. Namun, jika jumlahnya berlebihan dalam tubuh, ini justru bisa menyebabkan munculnya berbagai penyakit, seperti jantung koroner dan diabetes. Berikut ini adalah cara mudah menghitung kebutuhan kalori harian yang bisa aplikasikan di rumah sebagai acuan dalam menjalankan program diet. Rumus menghitung BBI (Berat Badan Ideal) adalah $BBI = 90\% (\text{tinggi badan} - 100)$ [7].

2.2. Augmented Reality

Realitas tertambah, atau kadang dikenal dengan singkatan bahasa Inggrisnya AR (*Augmented Reality*), adalah teknologi yang menggabungkan benda maya dua dimensi dan ataupun tiga dimensi ke dalam sebuah lingkungan nyata tiga dimensi lalu memproyeksikan benda – benda maya tersebut dalam waktu nyata (real time). Tidak seperti realitas maya (*virtual reality*) yang sepenuhnya menggantikan kenyataan, *augmented reality* sekedar menambahkan atau melengkapi kenyataan [8]. AR dapat diaplikasikan untuk semua indera, tidak hanya visual, termasuk pendengaran, sentuhan, dan penciuman. Gunanya untuk memperkaya pengalaman penggunaannya, membantu persepsi dan

interaksi penggunaanya dengan dunia nyata. Informasi yang ditampilkan oleh benda maya membantu pengguna melaksanakan kegiatan dalam dunia nyata. AR bisa digunakan dalam beberapa bidang penting seperti kesehatan, militer dan industri manufaktur. AR juga telah diaplikasikan dalam perangkat bergerak sudah digunakan oleh orang banyak, seperti telepon genggam [8].

Teknologi *augmented reality* sangat cepat sekali berkembang, di Indonesia sendiri telah banyak aplikasi – aplikasi yang menggunakan teknologi *augmented reality*. *Augmented reality* merupakan terobosan dibidang teknologi yang sangat canggih. Karena dengan teknologi ini kita dapat membuat segala hal yang abstrak atau virtual bisa kelihatan nyata atau real. Teknologi AR sendiri telah dikembangkan dalam berbagai hal, dalam pemanfaatanya teknologi ini dapat digunakan dalam hal *Interactive Games, Presentation, Event, High Tech Environment, Website, Promotion* [8]. Salah satu aplikasi yang memanfaatkan AR dapat dilihat pada Gambar 2.1.



Gambar 2.1. *Augmented Reality* dengan Objek Buah Apel

2.3. Sistem Operasi Android dan Android Studio

Sistem Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat seluler layar sentuh seperti telepon pintar dan komputer tablet. Android awalnya dikembangkan oleh Android, Inc., dengan dukungan finansial dari Google, yang kemudian membelinya pada tahun 2005. Sistem operasi ini dirilis secara resmi pada tahun 2007, bersamaan dengan didirikannya Open Handset Alliance, konsorsium dari perusahaan-perusahaan perangkat keras, perangkat lunak, dan telekomunikasi yang bertujuan untuk memajukan standar terbuka perangkat seluler. Ponsel Android pertama mulai dijual pada bulan Oktober 2008 [9].

Android Studio adalah aplikasi pengembangan baru dan terintegrasi penuh, yang baru saja dirilis oleh Google untuk sistem operasi Android. Android Studio dirancang untuk menjadi peralatan baru dalam pengembangan aplikasi dan juga memberi alternatif lain selain Eclipse yang saat ini menjadi IDE yang paling banyak dipakai. Saat memulai proyek baru dengan Android Studio, struktur proyek akan muncul bersama dengan hampir semua berkas yang ada di dalam direktori SDK, peralihan ke sistem manajemen berbasis Gradle ini memberikan fleksibilitas yang lebih besar pada proses pembangunannya. Android Studio memberikan kesempatan bagi pengguna untuk melihat perubahan visual apapun yang Anda lakukan pada aplikasi secara langsung. Anda juga bisa melihat perbedaannya jika dipasang pada beberapa perangkat Android berbeda, termasuk konfigurasi dan resolusinya secara bersamaan [10].

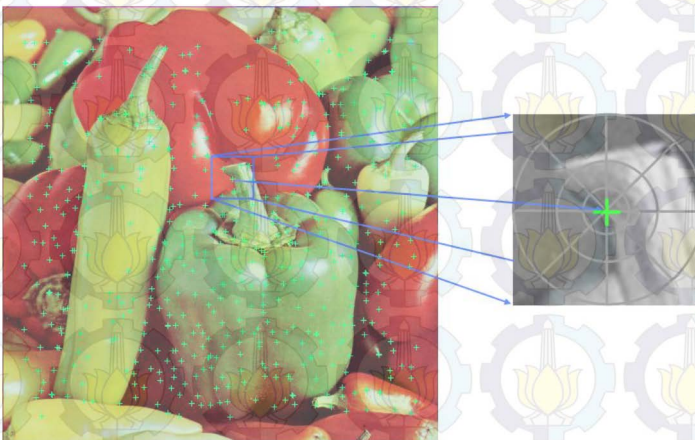
2.4. SURF (*Speeded-Up Robust Features*)

SURF (*Speeded-Up Robust Features*) merupakan sebuah algoritma yang cepat dan akurat untuk proses mendeteksi deskriptor lokal dari kesamaan representasi citra *invariant*. Deskriptor adalah sebuah ciri-ciri dari suatu citra berdasarkan

aturan tertentu dari suatu algoritma. SURF menggunakan citra *integral* untuk meningkatkan kecepatan komputasi. *Integral Image* adalah sebuah citra dimana nilai setiap piksel merupakan akumulasi dari piksel atas dan kirinya. Sebagai contoh, piksel (a,b) memiliki nilai akumulatif untuk semua piksel (x,y) dimana $x \leq a$ dan $y \leq b$. Algoritma ini didasarkan pada kerangka SURF dari hasil disertasi Herbert Bay [11]. Secara umum algoritma ini terdiri dari tiga bagian utama yaitu pendeteksian *keypoint*, pendeskripsian *keypoint*, dan pencocokan deskriptor.

2.4.1. Pendeteksian *Keypoint*

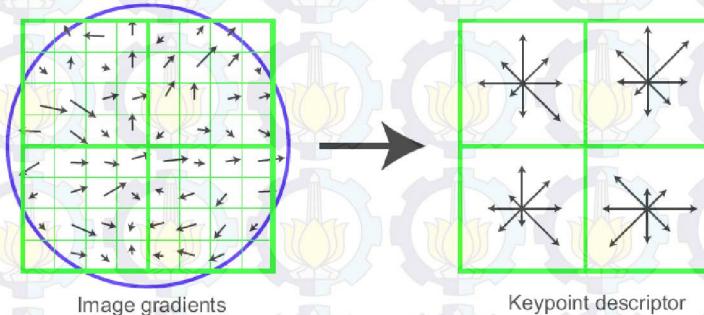
Pendeteksian *keypoint* didasarkan pada matrik Hessian. Tahap ini akan memberikan keluaran berupa *keypoint*. *Keypoint* merupakan *interest point* yang dianggap valid karena sudah berada dibawah batas nilai yang ditetapkan. Determinasi dari matrik Hessian digunakan untuk mendeterminasi lokasi dan skala untuk menyeleksi kandidat *interest point*. Kandidat *interest point* akan dianggap valid apabila mempunyai nilai dibawah batas nilai yang diberikan. Kemudian dilakukan perbaikan pada *box-space* [11]. Pendeteksian *keypoint* dapat dilihat pada Gambar 2.2.



Gambar 2.2. Pendeteksian *Keypoint Point*

2.4.2. Pendeskripsian *Keypoint*

Pendeskripsian *keypoint* dibagi menjadi dua tahap. Tahap pertama adalah menetapkan orientasi berdasarkan informasi dari daerah melingkar di sekitar *interest point* yang terdeteksi.



Gambar 2.3. Pendeskripsian *Keypoint*

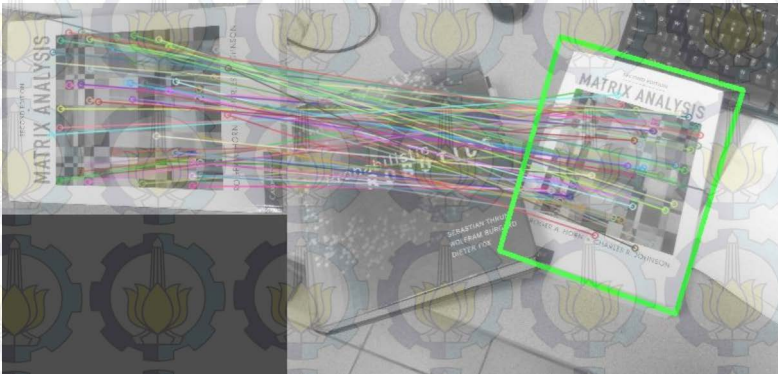
Tahap kedua adalah membuat *grid* sebesar 64 (8x8) yang digunakan untuk menampung deskriptor yang berkorespondensi dengan histogram [11]. Pendeskripsian *keypoint* dapat dilihat pada Gambar 2.3.

2.4.3. Pencocokan Deskriptor

Pencocokan deskriptor adalah pendeskripsian titik-titik fitur menjadi deskriptor vektor agar titik-titik fitur memiliki ketahanan terhadap rotasi, kontras, dan perubahan sudut pandang. Setiap fitur yang terdeteksi akan diberikan orientasi agar tahan terhadap rotasi. Vektor fitur deskriptor yang digunakan pada SURF ini berupa *keypoint* sejumlah 400 dengan *hessian-threshold* sejumlah 100. Kemudian dilakukan perbandingan kecocokan fitur menggunakan metode *Euclidean Distance* antara deskriptor gambar masukan dengan semua deskriptor gambar yang sudah dilatih pada direktori basis data gambar [11]. Perhitungan pencocokan deskriptor dengan metode *Euclidean Distance* dapat dilihat pada Persamaan 2.1

$$D(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_n - y_n)^2} \quad (2.1)$$

Pada Persamaan 2.1. dimana (x, y) adalah fitur deskriptor *keypoint* pada gambar. Sedangkan visualisasi pencocokan deskriptor gambar dapat dilihat pada Gambar 2.4.



Gambar 2.4. Visualisasi Pencocokan Deskriptor Gambar

2.5. OpenCV

OpenCV (*Open Source Computer Vision Library*) adalah sebuah pustaka perangkat lunak yang ditujukan untuk pengolahan citra dinamis secara *real-time*, yang pemngembangannya diawali oleh Intel, dan sekarang didukung oleh Willow Garage dan Itseez. OpenCV dirilis dibawah lisensi permisif BSD yang lebih bebas dari pada GPL, dan memberikan kebebasan sepenuhnya untuk dimanfaatkan secara komersil tanpa perlu mengungkapkan kode sumbernya. Ia juga memiliki antar muka yang mendukung bahasa pemrograman C++, C, Python dan Java, termasuk untuk sistem operasi Windows, Linux, Mac OS, iOS dan Android. OpenCV didisain untuk efisiensi dalam komputasi dan difokuskan pada aplikasi *real-time* [12].

Ditulis dalam bahasa C++, C, Python dan Java yang dioptimalkan, dengan perpustakaan dapat memanfaatkan pengolahan pada prosesor inti majemuk (*multi-core processing*). Mendukung OpenCL, sekaligus memberikan keuntungan dari akselerasi hardware yang dimiliki platform komputasi heterogen. OpenCV telah diadopsi di seluruh dunia, memiliki lebih dari 47 ribu komunitas pengguna dengan estimasi download melebihi 7 juta. Penggunaannya openCV disebutkan, mulai dari seni interaktif, meliputi inspeksi penambangan (*mines inspection*), *stitching maps* di *website*, sampai dengan robotika maju (*advanced*) [12].

2.6. Platform Fatsecret API

Platform *FatSecret* menyediakan informasi nutrisi dari makanan dan produk makanan non lokal yang dapat dipercaya, sehingga pengguna dapat melakukan pemilihan informasi nutrisi makanan apa saja yang akan dimakan. API *Fatsecret* menyediakan akses ratusan basis data nutrisi yang komprehensif bagi pengembang perangkat lunak dengan mengakses informasi gizi melalui REST (*Representational State Transfer*) dan *JavaScript*. API ini gratis untuk digunakan dan didokumentasikan dengan baik secara online beserta contoh kode program. Contoh kode program untuk mengakses basis data nutrisi *Fatsecret* dapat dilihat pada Kode Sumber 2.1.

```
consumer_key = 'consumer key'
consumer_secret = 'consumer secret'

fs = Fatsecret(consumer_key, consumer_secret)
foods = fs.foods_search()
```

Kode Sumber 2.1. Contoh Kode Program untuk Mengakses Basis Data Nutrisi *Fatsecret*

Untuk menggunakan API *Fatsecret*, pengembang harus melakukan registrasi pada *Fatsecret Platform API*. Setelah

registrasi, pengembang akan mendapatkan *REST API Consumer Key* dan *REST API Shared Secret* untuk mengakses API *Fatsecret*. API *Fatsecret* menggunakan OAuth (*Open Authentification*) untuk memverifikasi kebenaran permintaan dan memiliki batas 5000 API panggilan per hari [13]. Basis data *FatSecret* adalah sebagai penyimpanan informasi data nutrisi berupa kalori, karbohidrat, lemak, dan protein. Informasi nama objek yang didapat akan dikirim ke metode *FatSecret Search*, untuk mendapatkan kembali daftar informasi makanan beserta rincian informasi nutrisi.



Gambar 2.5. Contoh Penggunaan *Fatsecret* untuk Pencarian Nutrisi Buah Apel

Masukan dari *Fatsecret Search* berupa tipe data *string* berupa jenis makanan yang akan dicari. Hasil dari *Fatsecret Search* adalah rangkuman nutrisi untuk setiap takaran saji. Salah satu contoh penggunaan *Fatsecret* dapat dilihat pada Gambar 2.5.

2.7. Webservice

Webservice adalah suatu sistem perangkat lunak yang dirancang untuk mendukung interoperabilitas dan interaksi antar sistem pada suatu jaringan. *Webservice* digunakan sebagai suatu fasilitas yang disediakan oleh suatu *website* untuk menyediakan layanan (dalam bentuk informasi) kepada sistem lain, sehingga sistem lain dapat berinteraksi dengan sistem tersebut melalui

layanan – layanan (*service*) yang disediakan oleh suatu sistem yang menyediakan *webservice*. *Webservice* menyimpan data informasi dalam format XML, sehingga data ini dapat diakses oleh sistem lain walaupun berbeda platform, sistem operasi, maupun bahasa *compiler* [14].

2.8. Flask Python

Flask Python merupakan *microframework* berbasis Python untuk membuat *webservice*. Disebutkan sebagai *microframework* karena pengembang aplikasi pada umumnya hanya menggunakan satu atau dua fungsionalitas. Flask Python sudah menyediakan pustaka untuk mendukung ekstensi yang bisa digunakan untuk menambahkan fitur pada aplikasi. Ekstensi pada Flask Python digunakan untuk mengirim dan menerima objek yang digunakan untuk komunikasi antara aplikasi dan server [15]. Aplikasi *webservice* bisa dibangun secara minimalis menggunakan Flask Python karena menggunakan fungsionalitas yang hanya dibutuhkan oleh pengembang saja. Berbasis Python karena *microframework* ini dibangun dan dikembangkan menggunakan bahasa Python. Salah satu contoh penggunaan Flask Python bisa dilihat pada Kode Sumber 2.2.

```
from flask import Flask
app = Flask(__name__)

@app.route("/")
def hello():
    return "Hello World!"

if __name__ == "__main__":
    app.run()
```

Kode Sumber 2.2. Contoh Penggunaan Flask Python

BAB III

ANALISIS DAN PERANCANGAN SISTEM

Bab ini membahas tahap analisis permasalahan dan perancangan dari sistem yang akan dibangun. Analisis topik permasalahan yang diangkat dalam pengerjaan tugas akhir. Analisis kebutuhan mencantumkan kebutuhan-kebutuhan yang diperlukan perangkat lunak. Selanjutnya dibahas mengenai perancangan sistem yang dibuat.

3.1. Analisis Perangkat Lunak

Pada subbab ini akan dibahas mengenai analisa kebutuhan perangkat lunak yang dibangun. Hal yang akan dibahas meliputi deskripsi umum perangkat lunak, spesifikasi kebutuhan perangkat lunak, dan identifikasi pengguna.

3.1.1. Deskripsi Umum Perangkat Lunak

Aplikasi media informasi nutrisi pada makanan atau produk makanan menggunakan teknologi *augmented reality* berbasis Android dibangun dengan tujuan agar kita bisa lebih memerhatikan kandungan nutrisi makanan yang akan dikonsumsi terutama kalori, karbohidrat, lemak dan protein sesuai dengan kebutuhan setiap hari. Dengan teknologi *augmented reality* diharapkan bisa lebih menarik pengguna aplikasi serta bisa menjadi media informasi nutrisi yang informatif.

Aplikasi media informasi nutrisi pada makanan atau produk makanan pada Tugas Akhir ini, menggunakan teknologi *augmented reality* berupa informasi nutrisi. Informasi yang ditampilkan berupa kandungan nutrisi yaitu kalori, lemak, karbohidrat dan protein. Misalnya pengguna aplikasi akan mengonsumsi 1 buah pisang, 1 buah jeruk dan 5 buah anggur. Untuk mengetahui kandungan nutrisi di dalam makanan atau produk makanan tersebut, pengguna bisa mendapatkan informasi

dengan mengambil foto makanan atau produk makanan. Aplikasi akan menampilkan informasi nutrisi yang disimpan di basis data dan dikemas dalam bentuk *augmented reality*. Menggunakan metode klasifikasi gambar sehingga bisa mengidentifikasi makanan atau produk makanan dari foto untuk mendapatkan informasi nama objek. Informasi nama objek yang didapat akan dikirim ke *Fatsecret Search*, untuk mendapatkan kembali daftar informasi makanan beserta rincian informasi nutrisi.

3.1.2. Spesifikasi Kebutuhan Perangkat Lunak

Kebutuhan sistem yang dibangun pada tugas akhir ini melibatkan dua hal, yakni kebutuhan fungsional maupun kebutuhan non-fungsional, dimana masing-masing berhubungan dengan keberhasilan dalam pembuatan aplikasi tugas akhir ini.

3.1.2.1. Kebutuhan Fungsional Perangkat Lunak

Pada sistem ini, terdapat beberapa kebutuhan fungsional yang mendukung untuk jalannya aplikasi. Fungsi yang terdapat dalam aplikasi ini adalah sebagai berikut.

a) Pengenalan Objek Gambar Makanan

Aplikasi mampu melakukan pengenalan objek dengan mengambil gambar foto dari makanan atau produk makanan yang kemudian diklasifikasikan untuk mendapatkan pada kelas mana gambar tersebut memiliki kemiripan. Hal yang ditekankan pada fungsionalitas ini adalah bagaimana mengetahui nama objek makanan atau produk makanan yang difoto dengan kamera *android*.

b) Identifikasi Nutrisi

Setelah nama objek gambar makanan atau produk makanan ditemukan, kemudian aplikasi mengidentifikasi kandungan nutrisi berupa kalori, karbohidrat, protein dan kalori berdasarkan nama objek. Identifikasi nutrisi dengan mencari pada basis data di *webservice*.

c) Informasi Nutrisi dalam *Augmented Reality* Teks

Aplikasi ini menampilkan informasi nutrisi berupa kalori, karbohidrat, protein dan lemak terkandung di dalam makanan atau produk makanan kepada pengguna perangkat bergerak berbasis Android menggunakan teknologi *augmented reality* teks sehingga lebih menarik.

3.1.2.2. Kebutuhan Non-Fungsional

Pada sistem ini, terdapat beberapa kebutuhan non-fungsional yang mendukung dan menambah performa untuk jalannya aplikasi. Fungsi yang terdapat dalam aplikasi ini adalah sebagai berikut.

a) Terhubung dengan Jaringan Internet

Jaringan internet sangat dibutuhkan dalam penggunaan aplikasi ini yaitu untuk berkomunikasi dengan *webservice*. Apabila aplikasi tidak terhubung dengan jaringan internet yang sama, pengguna tidak bisa mendapatkan informasi nama makanan yang diidentifikasi beserta nutrisi yang terkandung didalamnya.

b) Didukung dengan CPU (*Central Processing Unit*)

CPU merupakan device terpenting dalam sebuah komputer karena merupakan otak dari komputer. CPU memiliki tugas utama yaitu untuk mengolah data berdasarkan instruksi dan data dari perangkat lunak. CPU digunakan untuk mendukung kecepatan komputasi pengenalan objek makanan.

3.1.3. Identifikasi Pengguna

Pengguna yang akan terlibat hanya terdapat satu orang saja, yakni orang yang menggunakan aplikasi. Pengguna akan berkomunikasi dengan *webservice* menggunakan komunikasi *client-server*.

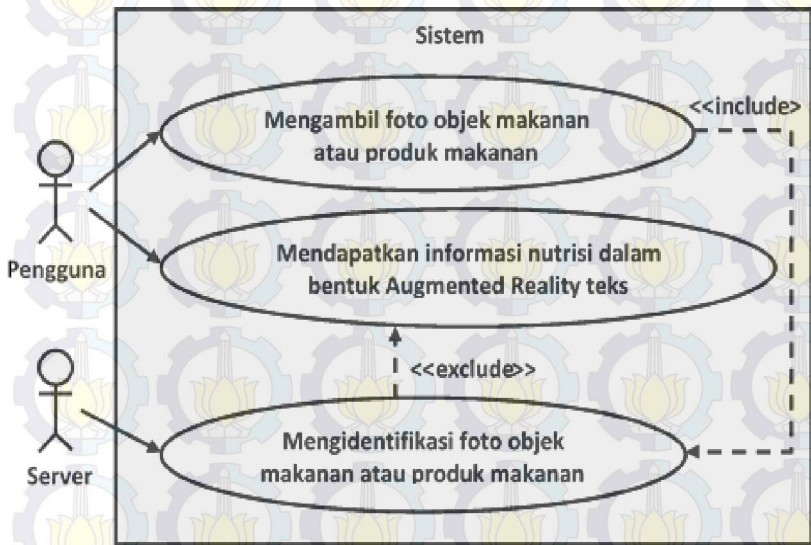
3.2. Perancangan Perangkat Lunak

Pada subbab ini membahas bagaimana rancangan dari aplikasi tugas akhir. Hal yang akan dibahas meliputi model kasus

penggunaan, definisi aktor, definisi kasus penggunaan, arsitektur umum sistem, rancangan antarmuka aplikasi, dan rancangan proses aplikasi.

3.2.1. Model Kasus Penggunaan

Dari hasil analisa deskripsi umum perangkat lunak dan spesifikasi kebutuhan yang telah dijelaskan, dapat digambarkan model kasus penggunaan aplikasi pada Gambar 3.1.



Gambar 3.1. Model Kasus Penggunaan Aplikasi

3.2.2. Definisi Aktor

Aktor yang terdapat pada aplikasi tugas akhir ini dapat dilihat pada Tabel 3.1.

Tabel 3.1. Definisi Aktor

No	Nama	Deskripsi
1	Pengguna	Merupakan aktor yang menggunakan aplikasi untuk mengambil dan

No	Nama	Deskripsi
		mengunggah foto objek makanan atau produk makanan ke server.
2	<i>Server</i>	Merupakan aktor dibalik berjalannya sistem yang mengomputasi foto objek makanan sehingga diperoleh informasi berupa nama dan nutrisi yang terkandung.

3.2.3. Definisi Kasus Penggunaan

Pada Gambar 3.1 telah dijelaskan bahwa aktor yang dalam hal ini disebut pengguna mempunyai tiga kasus penggunaan, yakni mengambil foto objek makanan atau produk makanan, mendapatkan informasi nutrisi dalam bentuk *augmented reality* teks, dan mengidentifikasi foto objek makanan atau produk makanan. Detail mengenai kasus penggunaan tersebut dapat dilihat pada Tabel 3.2.

Tabel 3.2. Definisi Kasus Penggunaan

No	Kode Kasus Penggunaan	Nama Kasus Penggunaan	Keterangan
1	KP-01	Mengambil foto objek makanan atau produk makanan.	Pengguna menggunakan makanan atau produk makanan sebagai objek yang akan difoto.
2	KP-02	Mendapatkan informasi nutrisi dalam bentuk <i>augmented reality</i> teks.	Pengguna mendapatkan rangkuman informasi nutrisi berupa kalori, karbohidrat, lemak, dan protein dalam bentuk <i>augmented reality</i> teks

No	Kode Kasus Penggunaan	Nama Kasus Penggunaan	Keterangan
			pada layar perangkat bergerak.
3	KP-03	Mengidentifikasi foto objek makanan atau produk makanan.	<i>Server</i> mengidentifikasi untuk mengenali nama objek yang difoto kemudian mencari pada basis data <i>fatsecret</i> mengenai nutrisi yang terkandung

3.2.3.1. Kasus Penggunaan Mengambil Foto Objek Makanan atau Produk Makanan

Spesifikasi kasus penggunaan mengambil foto objek makanan atau produk makanan dapat dilihat pada Tabel 3.3.

Tabel 3.3. Spesifikasi Kasus Penggunaan Mengambil Foto Objek Makanan atau Produk Makanan

Nama Kasus Penggunaan	Mengambil foto objek makanan atau produk makanan.
Nomor	KP-01
Deskripsi	Pengguna menggunakan makanan atau produk makanan sebagai objek untuk diambil gambar foto
Aktor	Pengguna
Kondisi Awal	Pengguna berada pada mode pengambilan foto di aplikasi dan terhubung dengan jaringan internet
Alur Normal	<ol style="list-style-type: none"> 1. Halaman utama muncul sebagai tanda aplikasi telah dijalankan. 2. Pengguna mengarahkan kamera ke objek makanan atau produk makanan.

	3. Pengguna menekan tombol <i>Capture Image</i> untuk mengambil foto objek 4. Foto objek dikirim ke <i>webservice</i> dengan ekstensi format berkas .jpg
Alur Alternatif	-
Kondisi Akhir	Foto objek terkirim ke <i>webservice</i>

3.2.3.2. Kasus Penggunaan Mendapatkan Informasi Nutrisi dalam Bentuk *Augmented Reality* Teks

Spesifikasi kasus penggunaan mendapatkan informasi nutrisi dalam bentuk *augmented reality* teks. dapat dilihat pada Tabel 3.4.

Tabel 3.4. Spesifikasi Kasus Penggunaan Mendapatkan Informasi Nutrisi dalam Bentuk *Augmented Reality* Teks

Nama Kasus Penggunaan	Mendapatkan informasi nutrisi dalam bentuk <i>augmented reality</i> teks.
Nomor	KP-02
Deskripsi	Pengguna dapat melihat informasi nutrisi di layar kamera berupa kalori, karbohidrat, lemak dan protein.
Aktor	Pengguna
Kondisi Awal	Informasi nama objek dan nutrisi terkandung dikembalikan ke perangkat bergerak pengguna
Alur Normal	1. Perangkat bergerak menerima informasi nama objek dan nutrisi terkandung.

	2. Perangkat menampilkan nama dan nutrisi terkandung di modus kamera aplikasi.
Alur Alternatif	A2. Nama objek makanan atau produk makanan salah 1. Pengguna mengambil foto objek kembali untuk diidentifikasi
Kondisi Akhir	Nama objek dan nutrisi berupa kalori, karbohidrat, lemak dan protein muncul di layar kamera pengguna

3.2.3.3. Kasus Penggunaan Mengidentifikasi Foto Objek Makanan atau Produk Makanan

Spesifikasi kasus penggunaan mengidentifikasi foto objek makanan atau produk makanan. dapat dilihat pada Tabel 3.5.

Tabel 3.5. Spesifikasi Kasus Penggunaan Mengidentifikasi Foto Objek Makanan atau Produk Makanan.

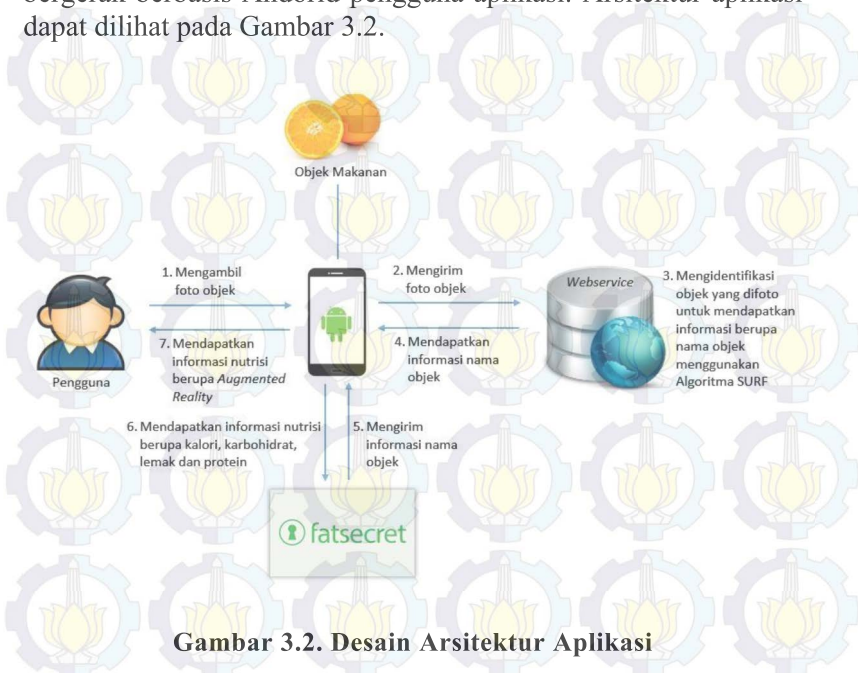
Nama Kasus Penggunaan	Mengidentifikasi foto objek makanan atau produk makanan.
Nomor	KP-03
Deskripsi	<i>Webservice</i> mengidentifikasi foto objek yang dikirim oleh aplikasi di perangkat bergerak pengguna
Aktor	<i>Webservice</i>
Kondisi Awal	<i>Webservice</i> menerima berkas dengan ekstensi .jpg dari perangkat bergerak pengguna
Alur Normal	1. <i>Webservice</i> menerima berkas dengan ekstensi .jpg 2. Ekstraksi fitur <i>keypoint</i> dari berkas .jpg

	3. Membandingkan fitur berkas .jpg dengan fitur yang disimpan di basis data. 4. Mengurutkan fitur basis data yang memiliki kemiripan pada fitur berkas .jpg berdasarkan kelas objek 5. Fitur di basis data yang memiliki kemiripan pada berkas .jpg adalah kelas objek dari berkas .jpg 6. Nama kelas objek dikirimkan ke API Platform Fatsecret 7. Fatsecret mencari kandungan nutrisi pada nama kelas objek berupa kalori, karbohidrat, protein dan lemak.
Alur Alternatif	-
Kondisi Akhir	Kandungan nutrisi berupa kalori, karbohidrat, protein dan lemak dikirimkan dalam ke perangkat bergerak pengguna

3.2.4. Arsitektur Umum Sistem

Aplikasi dibangun menggunakan arsitektur perangkat lunak yang mengintegrasikan perangkat bergerak dengan *webservice*. Dimana perangkat bergerak berbasis Android digunakan oleh pengguna aplikasi, kemudian modul kamera untuk pengambilan foto diarahkan pada objek makanan atau produk makanan. Foto objek akan dikirimkan ke *webservice*, dimana *webservice* mengidentifikasi untuk mendapatkan kembali informasi berupa nama objek. *Webservice* mengirimkan informasi berupa nama objek untuk mencari di basis data *Fatsecret* berupa informasi kalori, karbohidrat, protein dan lemak berdasarkan nama objek yang didapatkan. Informasi nutrisi berupa kalori, karbohidrat, protein dan lemak ditampilkan dalam bentuk *augmented reality*.

Rangkuman informasi nutrisi ditampilkan pada layar perangkat bergerak berbasis Andorid pengguna aplikasi. Arsitektur aplikasi dapat dilihat pada Gambar 3.2.



Gambar 3.2. Desain Arsitektur Aplikasi

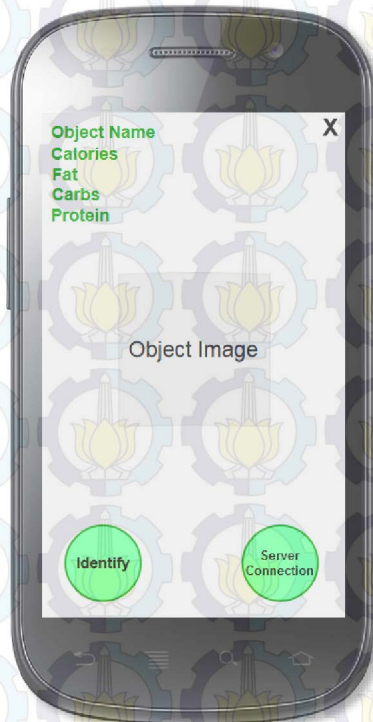
3.2.5. Rancangan Antarmuka Aplikasi

Rancangan antarmuka aplikasi diperlukan untuk memberikan gambaran umum kepada pengguna mengenai bagaimana sistem yang ada dalam aplikasi ini berinteraksi dengan pengguna.

3.2.5.1. Rancangan Antarmuka Halaman Utama

Pada halaman utama aplikasi akan menampilkan modus kamera yang siap untuk pengambilan gambar foto objek makanan atau produk makanan. Pada halaman ini terdapat tombol *Identify* untuk mengambil gambar objek makanan serta menampilkan kandungan nutrisi makanan. Tombol *Server Connection* berfungsi

untuk mengatur Alamat IP pada jaringan yang terhubung dengan aplikasi. Rancangan antarmuka halaman utama dilihat pada Gambar 3.3.

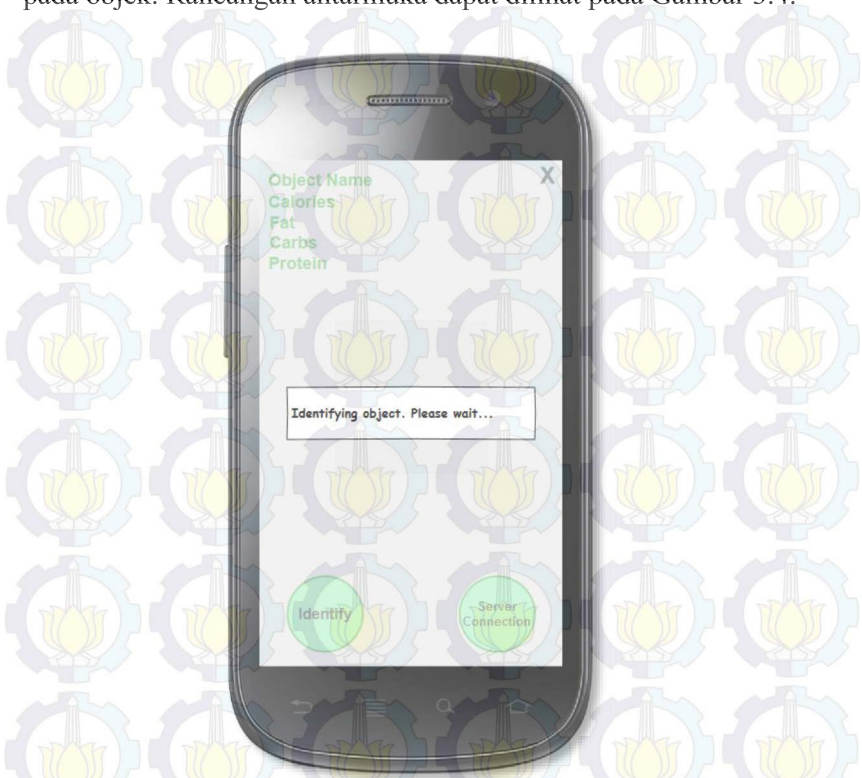


Gambar 3.3. Rancangan Antarmuka Halaman Utama

3.2.5.2. Rancangan Antarmuka Pengiriman Gambar ke Webservice

Rancangan antarmuka pengiriman gambar ke *webservice* akan menampilkan pesan peringatan bahwa gambar yang telah diambil sedang dikirim ke *server* untuk diidentifikasi. Gambar objek yang dikirimkan berupa berkas dengan ekstensi *.jpg*. Pengiriman gambar ke *server* berguna untuk mendapatkan nama

dari objek yang difoto dan rangkuman nutrisi yang terkandung pada objek. Rancangan antarmuka dapat dilihat pada Gambar 3.4.



Gambar 3.4. Rancangan Antarmuka Pengiriman Gambar ke Server

3.2.5.3. Rancangan Antarmuka Rangkuman Nutrisi

Pada halaman ini aplikasi akan menampilkan rangkuman nutrisi pada layar modus kamera di antarmuka aplikasi. Tombol *Identify* berguna untuk mengambil gambar objek serta menampilkan rangkuman nutrisi yang terkandung pada objek makanan yang difoto oleh pengguna. Rangkuman nutrisi

didapatkan dari hasil pencarian di *server* basis data *Fatsecret*. Rangkuman nutrisi berupa nama objek makanan beserta jumlah takaran, kalori, lemak, karbohidrat dan protein. Rancangan antarmuka dapat dilihat pada Gambar 3.5.



Gambar 3.5. Rancangan Antarmuka Rangkuman Nutrisi

3.2.5.4. Rancangan Antarmuka *Server Connection*

Rancangan antarmuka *server connection* berguna sebagai pengaturan koneksi jaringan internet yang terhubung dengan aplikasi dan *webservice*. Aplikasi dan *webservice* harus terhubung dalam satu jaringan koneksi yang sama. Terdapat tombol *server*

address untuk mengakses halaman permukaan untuk memasukkan alamat IP *server*. Rancangan antarmuka dapat dilihat pada Gambar 3.6.



Gambar 3.6. Rancangan Antarmuka *Server Connection*

3.2.5.5. Rancangan Antarmuka Masukkan Alamat IP *Server*

Pada rancangan antar muka ini berfungsi untuk memasukkan alamat IP yang terhubung pada aplikasi. Alamat IP *server* yang dimasukkan tergantung pada koneksi yang terhubung antara aplikasi dan *webservice*. Antar muka ini memberikan manfaat kepada pengguna agar bisa dinamis mengganti alamat IP *server*. Dengan mengetikkan sendiri pada alamat IP *server*, pengguna bisa

memasukkan langsung alamat IP *server* yang terhubung antara aplikasi dan *webservice*. Rancangan antarmuka dapat dilihat pada Gambar 3.7.

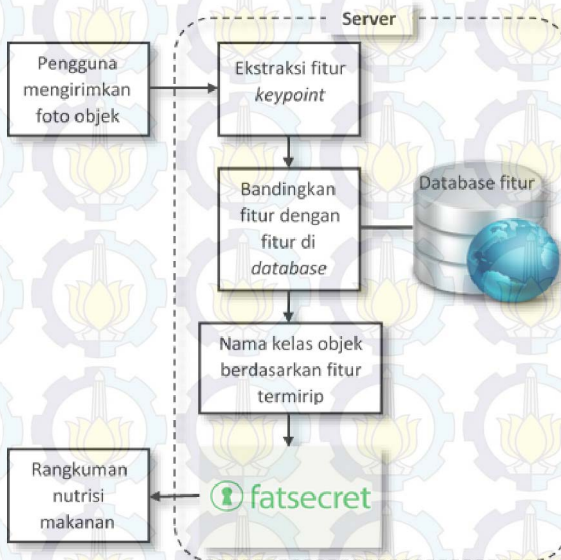


Gambar 3.7. Rancangan Antarmuka Masukkan Alamat IP Server

3.2.5.6. Rancangan Antarmuka *Webservice*

Rancangan antarmuka *webservice* berguna untuk mendapatkan rangkuman nutrisi berdasarkan foto objek yang diidentifikasi. *Webservice* menerima masukan berupa gambar. Kemudian gambar diekstraksi fitur berdasarkan fitur keypoint. Lalu fitur gambar dibandingkan dengan fitur di basis data.

Selanjutnya nama kelas objek berdasarkan fitur termirip dikirim ke *Fatsecret* untuk mengetahui kandungan nutrisi yang terkandung pada objek gambar. Fatsecret sudah terdapat basis data makanan beserta kandungan nutrisinya. Setelah itu rangkuman nutrisi berupa kalori, lemak, karbohidrat, dan protein dikirimkan ke perangkat bergerak pengguna. Rancangan antarmuka dapat dilihat pada Gambar 3.8.



Gambar 3.8. Rancangan Antarmuka Webservice

3.2.6. Rancangan Proses Penggunaan Aplikasi

Dalam proses ini terdapat proses dimana pengguna menggunakan aplikasi hingga mendapatkan informasi nutrisi yang terkandung di dalam objek makanan. Pengguna mengirimkan foto objek makanan dengan ukuran 400 x 400 piksel. Pada lapisan *server* terdapat basis data gambar untuk masing-masing kelas sejumlah 20 gambar. Gambar pada basis data, masing-masing berukuran 400 x 400 piksel.



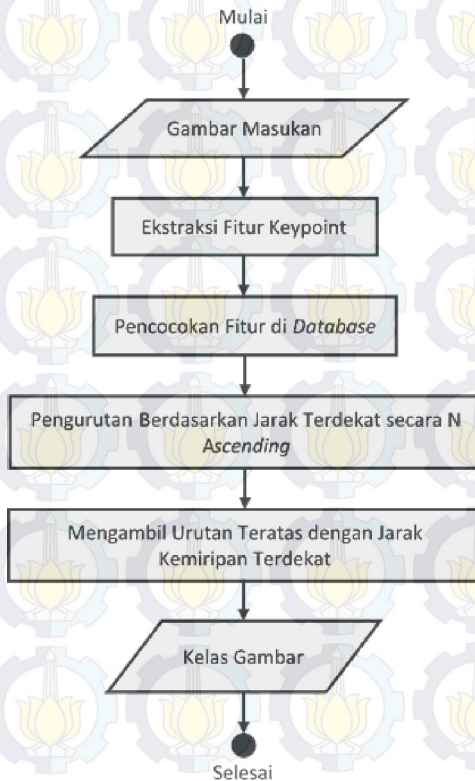
Gambar 3.9. Diagram Alir Proses Penggunaan Aplikasi

Untuk mendapatkan fitur gambar, ekstraksi fitur menggunakan pustaka SURF (*Speeded-up Robust Features*) yang sudah disediakan oleh OpenCV. Setelah didapatkan fitur gambar

berupa *keypoint*, kemudian dihitung kemiripan fitur menggunakan *Euclidian Distance* dengan Persamaan 2.1. Rancangan proses penggunaan aplikasi dapat dilihat pada Gambar 3.9.

3.2.7. Rancangan Proses Klasifikasi Gambar

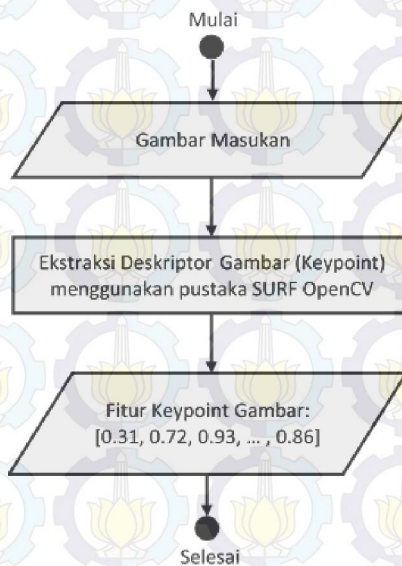
Proses ini sangat dibutuhkan dalam berjalannya aplikasi. Proses klasifikasi gambar berguna untuk menentukan gambar termirip di basis data gambar masukan yang dikirimkan oleh perangkat bergerak. Secara umum proses klasifikasi gambar dapat dilihat pada Gambar 3.10.



Gambar 3.10. Diagram Alir Proses Klasifikasi Gambar

3.2.7.1. Rancangan Proses Ekstraksi Fitur *Keypoint* Gambar

Proses ini sangat dibutuhkan dalam berjalannya aplikasi, karena pada proses ekstraksi fitur *keypoint* terdapat pengambilan deskriptor yang diperlukan untuk menentukan kelas objek gambar atau citra di basis data. Proses ekstraksi fitur menggunakan pustaka SURF (*Speeded-Up Robust Features*) yang sudah disediakan oleh OpenCV. Citra gambar masukan yang dikirim dari Android ke *Webservice* akan melalui proses pengambilan fitur deskriptor. Fitur deskriptor yang diambil berupa *keypoint* yang kemudian disimpan dalam berkas berekstensi .pkl. Rancangan proses ekstraksi fitur *keypoint* gambar bisa dilihat pada Gambar 3.11.



Gambar 3.11. Proses Ekstraksi Fitur *Keypoint* Gambar

3.2.7.2. Rancangan Proses Perhitungan Kemiripan Gambar

Proses perhitungan kemiripan gambar sangat dibutuhkan dalam menentukan kelas objek gambar. Dilakukan pendeskripsian

titik-titik fitur menjadi deskriptor vektor agar titik-titik fitur memiliki ketahanan terhadap rotasi, kontras, dan perubahan sudut pandang. Setiap fitur yang terdeteksi akan diberikan orientasi agar tahan terhadap rotasi. Fitur vektor deskriptor yang digunakan pada SURF (*Speeded-Up Robust Features*) ini berupa *keypoint* sejumlah 400 dengan *hessian-threshold* sejumlah 100. *Hessian-threshold* adalah ambang batas fitur pengambilan *keypoint* untuk perhitungan kemiripan. *Keypoint* sebagai fitur deskriptor dapat dihitung kemiripannya dengan menggunakan fungsi *Euclidian Distance* tanpa dilakukan normalisasi fitur. Perhitungan menggunakan *Euclidian Distance* dapat dilihat pada Persamaan 2.1. Semakin kecil jarak *Euclidian Distance* maka semakin besar kemiripan fitur gambar. Keluaran pada proses perhitungan kemiripan gambar ini berupa nama objek makanan atau produk makanan. Proses perhitungan kemiripan gambar bisa dilihat pada Gambar 3.12.



Gambar 3.12. Proses Perhitungan Kemiripan Gambar

BAB IV

IMPLEMENTASI

Bab ini membahas tentang implementasi dari perancangan sistem. Bab ini berisi proses implementasi dari setiap perancangan pada bab sebelumnya. Namun, pada hasil akhir mungkin saja terjadi perubahan kecil. Implementasi pembangunan *webservice* menggunakan bahasa Python 2.7 dan untuk *client* Android menggunakan Android Studio.

4.1. Lingkungan Pembangunan

Dalam pembangunan aplikasi, digunakan beberapa perangkat sebagai pendukung baik perangkat keras maupun perangkat lunak. Lingkungan pembangunan dijelaskan sebagai berikut.

4.1.1. Lingkungan Pembangunan Perangkat Keras

Perangkat keras yang digunakan dalam pembuatan aplikasi ini adalah sebuah laptop Toshiba Satellite L745 14-inch dan perangkat bergerak sebagai alat uji coba dengan spesifikasi sebagai berikut.

- Prosesor Intel(R) Core(TM) i5 - 2410M CPU @ 2,30GHz
- (RAM) 4,00 GB
- NVIDIA GeForce GT 525M
- Asus Zenfone 4 Android OS 4.3 (Jelly Bean) RAM 1GB Camera 5MP

4.1.2. Lingkungan Pembangunan Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam pembuatan aplikasi adalah sebagai berikut.

- Python 2.7
- Sublime Text 3
- Android Studio 1.2
- Windows 7 Professional 64-bit

4.2. Implementasi Antarmuka Aplikasi Android

Pada subbab ini akan dibahas mengenai hasil implementasi yang dilakukan berdasarkan rancangan antarmuka. Nantinya akan digunakan kamera perangkat bergerak Android untuk pengambilan gambar objek makanan atau produk makanan yang kemudian diproses oleh *webservice* untuk mendapatkan kandungan nutrisi. Implementasi Antarmuka pada Android menggunakan Android Studio, sedangkan *webservice* menggunakan Sublime Text 3 menggunakan bahasa pemrograman Python 2.7.

4.2.1. Implementasi Antarmuka Halaman Utama

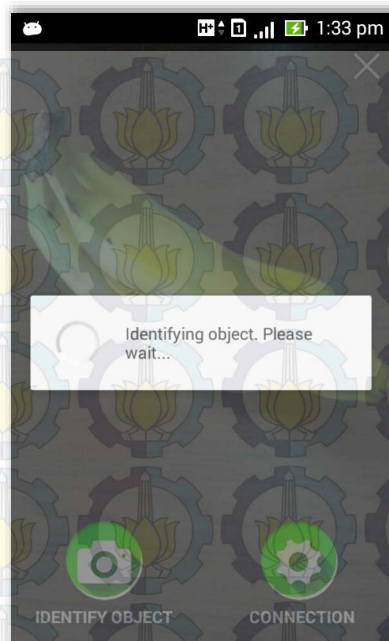
Antarmuka Halaman Utama adalah antarmuka yang pertama kali akan muncul pada saat pengguna membuka aplikasi. Antarmuka ini berguna untuk mengambil gambar dari objek makanan atau produk makanan. Pertama kali aplikasi dijalankan maka akan berada pada modus pengambilan gambar objek. Terdapat tombol *identify object* untuk mengambil gambar objek makanan serta menampilkan kandungan nutrisi makanan. Tombol *connection* berfungsi untuk mengatur alamat IP pada jaringan yang terhubung dengan aplikasi. Pada layar bagian kanan atas terdapat tombol untuk keluar dari aplikasi. Implementasi dari antarmuka dapat dilihat pada Gambar 4.1.



Gambar 4.1. Antarmuka Halaman Utama

4.2.2. Implementasi Antarmuka Pengiriman Gambar ke Webservice

Antarmuka pengiriman gambar ke *webservice* adalah sebuah antarmuka yang disediakan aplikasi untuk menampilkan pesan peringatan bahwa gambar yang telah diambil sedang dikirim ke *server* untuk diidentifikasi. Gambar objek makanan atau produk makanan yang dikirimkan berupa berkas dengan ekstensi *.jpg*. Setelah *server* menerima gambar yang dikirimkan oleh Android, gambar diidentifikasi untuk mendapatkan nama dan nutrisi yang terkandung pada makanan atau produk makanan. Implementasi dari antarmuka dapat dilihat pada Gambar 4.2.



Gambar 4.2. Antarmuka Pengiriman Gambar ke *Webservice*

4.2.3. Implementasi Antarmuka Rangkuman Nutrisi

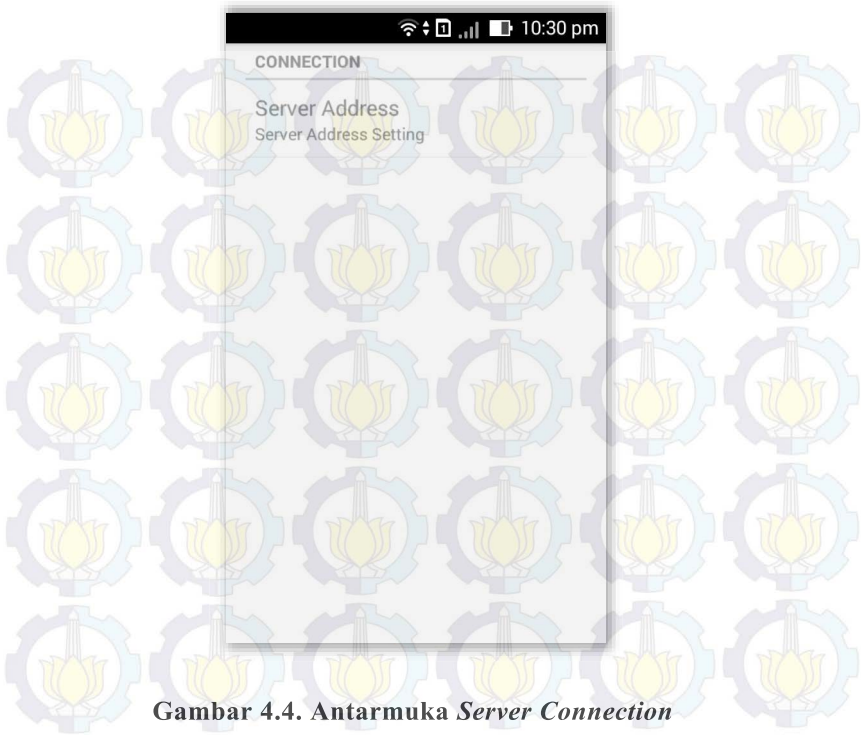
Antarmuka rangkuman nutrisi akan menampilkan informasi nutrisi pada layar modus kamera di antarmuka aplikasi. Apabila ingin mengidentifikasi objek kembali, pengguna bisa menggunakan tombol *identify object* pada layar kamera. Rangkuman nutrisi didapatkan dari hasil pencarian di *server* basis data *Fatsecret*. Rangkuman nutrisi berupa nama objek makanan beserta jumlah takaran untuk setiap konsumsi, kalori, lemak, karbohidrat dan protein.. Implementasi dari antarmuka dapat dilihat pada Gambar 4.3.



Gambar 4.3. Antarmuka Rangkuman Nutrisi

4.2.4. Implementasi Antarmuka *Server Connection*

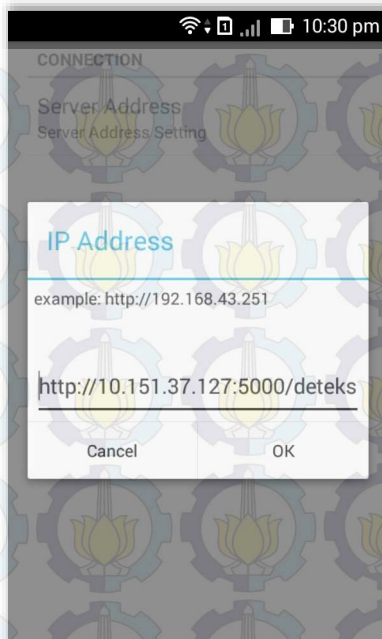
Antarmuka *server connection* menampilkan tampilan berupa layar aktifitas untuk pengaturan alamat IP *server*. Terdapat teks *connection* sebagai nama dari aktifitas layar. Dan tombol *server address* untuk menampilkan halaman permukaan memasukkan alamat IP *server*. Implementasi antarmuka dapat dilihat pada Gambar 4.4.



Gambar 4.4. Antarmuka *Server Connection*

4.2.5. Implementasi Antarmuka Masukkan Alamat IP *Server*

Antarmuka masukkan alamat IP *server* berfungsi untuk menghubungkan antara aplikasi dengan *webservice*. Aplikasi harus terhubung dengan jaringan internet yang sama sehingga konfigurasi alamat IP *server* yang terhubung antar keduanya menggunakan jaringan lokal yang sama. Implementasi antar muka dapat dilihat pada Gambar 4.5.



Gambar 4.5. Antarmuka Masukkan Alamat IP Server

4.3. Implementasi Antarmuka *Webservice*

Pada subbab ini akan dibahas mengenai implementasi aplikasi pada bagian *webservice*. Dijelaskan juga mengenai fungsi – fungsi yang dibutuhkan untuk menunjang aplikasi agar dapat berjalan sebagaimana mestinya. Implementasi ini dilakukan menggunakan Sublime Text 3 dengan bahasa pemrograman Python 2.7.

4.3.1. Implementasi Lapisan *Server* pada *Webservice*

Implementasi *webservice* menggunakan pustaka Flask Python dengan metode POST dan GET. Metode POST dan GET digunakan untuk komunikasi pengiriman data berupa gambar

objek makanan antara Android dan *webservice*. Ketika Android mengirimkan file berupa gambar objek makanan, *webservice* akan menyimpan file pada direktori *testing* yang kemudian diekstraksi fitur dalam berkas .pkl. Ekstraksi fitur yang disimpan akan diproses untuk dihitung kemiripan dengan berkas .pkl pada direktori basis data. Untuk lebih detailnya dapat dilihat pada Kode Sumber 4.1.

```
app = Flask(__name__)
app.debug = True

@app.route("/deteksi", methods=['GET', 'POST'])
def deteksi():
    if request.method == 'POST':
        file = request.files['file']
        file.save('testing/' +
file.filename)
        (kelas,info) =
sim.deteksi_image(file.filename)
        return str(kelas)+"#"+str(info)

if __name__ == "__main__":
    app.run(host = '0.0.0.0', debug = True)
```

Kode Sumber 4.1. Implementasi Server pada Webservice

Koneksi Android dan *webservice* menggunakan *localhost connection* sehingga aplikasi yang digunakan harus terhubung jaringan lokal untuk komunikasi.

4.3.1.1. Ekstraksi Fitur *Keypoint*

Pengambilan fitur pada gambar tidak menggunakan ekstraksi fitur primitif seperti warna, bentuk, luasan dan lain sebagainya. Ekstraksi fitur yang diambil berupa *keypoint* yaitu *interest point* pada kandidat piksel yang menyimpan piksel terunik sebagai ciri khas objek gambar. *Keypoint* kemudian dijadikan sebagai deskriptor untuk proses pencocokan gambar. Proses ekstraksi fitur ini menggunakan pustaka OpenCV Python. Untuk lebih detailnya dapat dilihat pada Kode Sumber 4.2.


```

def save_feature(keypoints, descriptors,
filename):
    i = 0
    temp_array = []
    for point in keypoints:
        temp = (point.pt, point.size,
point.angle, point.response, point.octave,
point.class_id, descriptors[i])
        i += 1
        temp_array.append(temp)
    s = pickle.dumps(temp_array)
    open(filename, 'w').write(s)

```

Kode Sumber 4.2. Implementasi Ekstraksi Fitur *Keypoint*

4.3.1.2. Menyimpan Hasil Ekstraksi Fitur

Pada fungsi menyimpan hasil ekstraksi fitur dilakukan pengambilan *keypoint* sebagai deskriptor untuk semua gambar di basis data direktori. Ambang batas deskriptor gambar menggunakan *hessian_threshold* dengan ketentuan apabila ambang semakin kecil maka deskriptor yang diambil akan banyak. Sebaliknya, apabila ambang semakin besar maka deskriptor yang diambil semakin sedikit. Pengambilan ekstraksi fitur menggunakan SURF OpenCV. Hasil ekstraksi fitur kemudian disimpan dalam berkas .pkl. Untuk lebih detailnya dapat dilihat pada Kode Sumber 4.3.

```

def comp_feature():
    images_train = [f for f in
os.listdir('.') if '.jpg' in f]
    hessian_threshold = 100
    detector = cv2.SURF(hessian_threshold)
    for image in images_train:
        img = cv2.imread(image)
        img_gray = cv2.cvtColor(img,
cv2.COLOR_BGR2GRAY)

```

```

        keypoints, desc =
detector.detectAndCompute(img_gray, None)
        save_feature(keypoints, desc,
image[:-4] + '.pkl')

```

Kode Sumber 4.3. Implementasi Penyimpanan Hasil Ekstraksi Fitur

4.3.1.3. Membaca Ekstraksi Fitur Gambar dari Android Untuk Dikomputasi

Pada fungsi ini terdapat proses pembacaan hasil ekstraksi gambar yang dikirim ke *webservice*. Hasil ekstraksi berupa deskriptor yang digunakan sebagai pencocokan gambar pada basis data di direktori. Pembacaan ekstraksi fitur mengambil vektor piksel gambar. Untuk gambar di basis data dikelompokkan berdasarkan nama objek sehingga hasil ekstraksi fitur berupa deskriptor yang tersimpan berdasarkan nama gambar. Deskriptor yang diambil disimpan ke berkas sementara yang kemudian dilakukan pencocokan di basis data direktori. Untuk lebih detail dapat dilihat pada Kode Sumber 4.4.

```

def load_feature(filename):
    features = pickle.loads(open(filename,
'r').read())
    keypoints, descriptors = [], []
    for feature in features:
        temp_feature =
cv2.KeyPoint(x=feature[0][0], y=feature[0][1],
_size=feature[1], _angle=feature[2],
_response=feature[3], _octave=feature[4],
_class_id=feature[5])
        temp_descriptor = feature[6]
        keypoints.append(temp_feature)
        descriptors.append(temp_descriptor)
    return keypoints, np.array(descriptors)

def comp_feature():
    images_train = [f for f in
os.listdir('.') if '.jpg' in f]

```

```

hessian_threshold = 100
detector = cv2.SURF(hessian_threshold)
for image in images_train:
    img = cv2.imread(image)
    img_gray = cv2.cvtColor(img,
cv2.COLOR_BGR2GRAY)
    keypoints, desc =
detector.detectAndCompute(img_gray, None)
    save_feature(keypoints, desc,
image[:-4] + '.pkl')

```

Kode Sumber 4.4. Implementasi Pembacaan Hasil Ekstraksi Fitur Gambar dari Android Untuk Dikomputasi

4.3.1.4. Mencocokkan Gambar di Basis Data Direktori

Pada fungsi ini terdapat proses pencocokan gambar yang dikirim dari Android ke *webservice*. Gambar yang dikirim dari Android sebelumnya diambil ekstraksi fitur terlebih dahulu kemudian dikomputasi dengan fitur – fitur setiap kelas objek di basis data direktori. Untuk setiap fitur gambar di basis data direktori dihitung nilai kemiripannya dengan gambar yang dikirim oleh Android. Semakin tinggi nilai kemiripan maka akan semakin tinggi juga untuk menjadi kandidat gambar termirip. Untuk lebih detail dapat dilihat pada Kode Sumber 4.5.

```

def deteksi_image(filename):
    images_train = [f for f in
os.listdir('.') if '.pkl' in f]
    hessian_threshold = 100
    detector = cv2.SURF(hessian_threshold)
    img = cv2.imread('testing/' + filename)
    img_gray = cv2.cvtColor(img,
cv2.COLOR_BGR2GRAY)
    keypoints, desc =
detector.detectAndCompute(img_gray, None)
    save_feature(keypoints, desc, 'testing/'
+ filename[:-4] + '.pkl')
    similarity = 0

```

```

        for image2 in images_train:
            val = compute_score2(
                'testing/' + filename[:-4] + '.pkl', image2)
            if similarity < val:
                similarity = val
                class_name = image2
            print
            image2,similarity
        print(class_name[:-6])

```

Kode Sumber 4.5. Implementasi Pencocokan Gambar di Basis data Direktori

4.3.1.5. Mendapatkan Rangkuman Nutrisi Objek Makanan

Pada fungsi ini terdapat *consumer key* dan *consumer secret* untuk menggunakan REST API pada *Fatsecret*. Untuk mendapatkan rangkuman nutrisi berupa kalori, lemak, karbohidrat dan protein menggunakan *food search*. Masukan dari *Fatsecret* ini berupa hasil klasifikasi gambar. Klasifikasi gambar dilakukan untuk mendapatkan nama objek gambar. Nama objek gambar tersebut yang kemudian digunakan sebagai kata kunci pencarian nutrisi di *Fatsecret*. Keluaran dari *Fatsecret* berupa kandungan nutrisi kalori, lemak, karbohidrat dan protein untuk setiap takaran penyajian. Untuk lebih detail dapat dilihat pada Kode Sumber 4.6.

```

        consumer_key='e0a7c995c2cf4e49973d316e4f
032a67'
        consumer_secret='8ac02431-eccb-40f5-
bf89-66ee7c765e51'
        fs=Fatsecret(consumer_key,consumer_secre
t)
        foods = fs.foods_search(class_name[:-6])
        print foods[0]['food_description']
        return(class_name[:-
6],foods[0]['food_description'])

```

Kode Sumber 4.6. Implementasi Mendapatkan Rangkuman Nutrisi Objek Makanan

BAB V

PENGUJIAN DAN EVALUASI

Bab ini membahas pengujian dan evaluasi pada aplikasi yang dikembangkan. Pengujian yang dilakukan adalah pengujian terhadap kebutuhan fungsional secara keseluruhan. Pengujian ini mengacu pada kasus penggunaan pada bab tiga. Pengujian dilakukan dengan beberapa skenario. Hasil evaluasi menjabarkan tentang rangkuman hasil pengujian pada bagian akhir bab ini.

5.1. Lingkungan Pembangunan

Dalam membangun aplikasi ini digunakan beberapa perangkat pendukung baik perangkat keras maupun perangkat lunak. Perangkat keras yang digunakan dalam pembuatan aplikasi ini adalah sebuah laptop Toshiba Satellite L745 14-inch dan perangkat bergerak sebagai alat uji coba dengan spesifikasi sebagai berikut.

- Prosesor Intel(R) Core(TM) i5 - 2410M CPU @ 2,30GHz
- (RAM) 4,00 GB
- NVIDIA GeForce GT 525M
- Asus Zenfone 4 Android OS 4.3 (Jelly Bean) RAM 1GB Camera 5MP

5.2. Skenario Pengujian

Skenario pengujian yang dilakukan adalah sebagai berikut.

1. Pengujian skenario 1 berupa pengujian terhadap kemampuan modul kamera dalam posisi siap untuk pengambilan gambar objek makanan.
2. Pengujian skenario 2 berupa pengujian terhadap penggunaan fungsionalitas pada tombol di layar modul kamera.

3. Pengujian skenario 3 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada buah pisang.
4. Pengujian skenario 4 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada buah jeruk.
5. Pengujian skenario 5 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada buah apel.
6. Pengujian skenario 6 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Indomie.
7. Pengujian skenario 7 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Pringles.
8. Pengujian skenario 8 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada buah Lemon .
9. Pengujian skenario 9 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Pocky.
10. Pengujian skenario 10 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Oreo.
11. Pengujian skenario 11 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Snickers.
12. Pengujian skenario 12 berupa pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Tango.
13. Pengujian skenario 13 berupa kalkulasi kebenaran aplikasi dalam mengidentifikasi nama dan nutrisi pada objek makanan atau produk makanan.
14. Pengujian skenario 14 berupa pengujian terhadap pengguna secara langsung yaitu dengan melibatkan

Mahasiswa Teknik Informatika dan kalangan masyarakat sekitar ITS sebagai pengguna langsung aplikasi.

5.2.1. Pengujian Skenario 1 dan Evaluasi

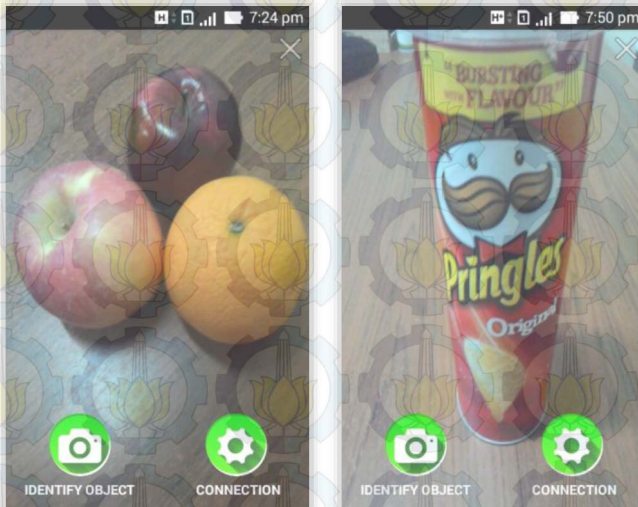
Pada pengujian skenario 1 ini akan dilakukan pengujian terhadap modus pengambilan gambar objek makanan atau produk makanan. Modus pengambilan gambar merupakan halaman utama yang akan keluar ketika aplikasi dijalankan. Pada modus kamera Terdapat dua tombol utama dalam menjalankan fungsionalitas aplikasi yaitu tombol *identify object* dan tombol *connection*. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.1.

Tabel 5.1. Skenario Pengujian 1

Nama Skenario Pengujian	Pengujian Modus Pengambilan Gambar
Kode	SP-01
Tujuan Pengujian	Menguji modus pengambilan gambar makanan atau produk makanan untuk diidentifikasi.
Kondisi Awal	Aplikasi terhubung pada jaringan internet yang sama dengan <i>webservice</i>
Data Input	-
Prosedur Pengujian	1. Siapkan perangkat bergerak yang sudah terpasang aplikasi. 2. Jalankan <i>webservice</i> pada perangkat keras di laptop.
Hasil yang Diharapkan	Setelah aplikasi dijalankan, halaman utama yaitu modus kamera dalam kondisi siap untuk mengambil gambar.

Hasil yang Diperoleh	Halaman utama yaitu modus kamera dalam kondisi siap untuk mengambil gambar.
----------------------	---

Pada awal pengujian, jalankan aplikasi pada perangkat bergerak. Letakan objek makanan atau produk makanan yang akan diidentifikasi. Arahkan modus kamera pada objek makanan atau produk makanan. Modus kamera pada aplikasi siap untuk proses pengambilan gambar. Terdapat tombol *identify object* untuk identifikasi gambar yang diambil pada objek dan tombol *connection* untuk pengaturan koneksi alamat IP jaringan internet yang terhubung. Proses pengujian terhadap modus pengambilan gambar objek makanan atau produk makanan dapat dilihat pada Gambar 5.1.



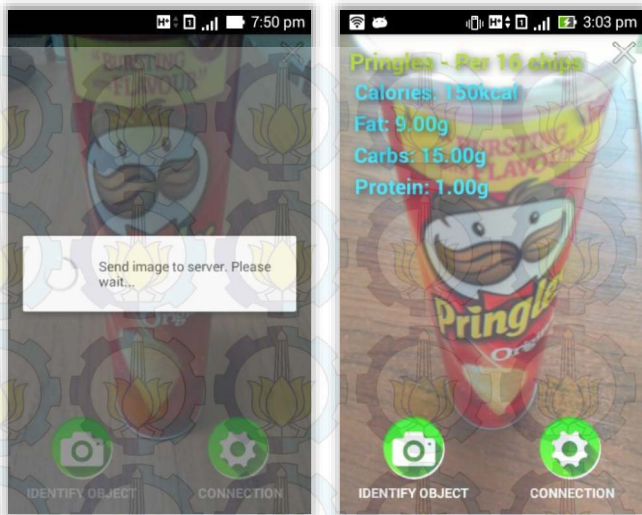
Gambar 5.1. Modus Pengambilan Gambar Objek Makanan atau Produk Makanan

5.2.2. Pengujian Skenario 2 dan Evaluasi

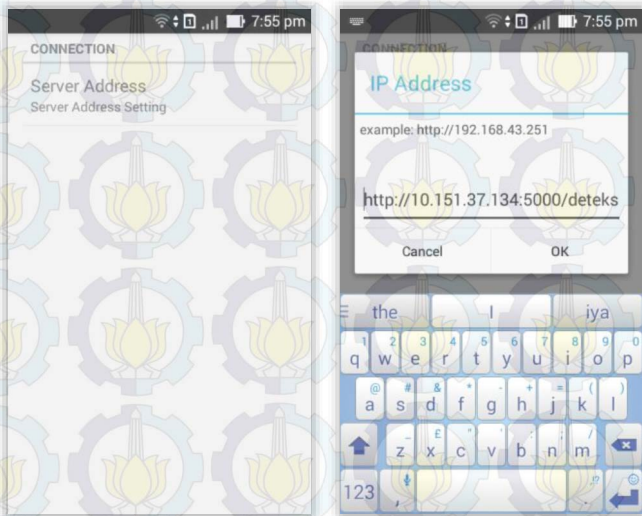
Pada pengujian skenario 2 ini akan dilakukan pengujian terhadap penggunaan fungsionalitas pada tombol di layar modus kamera. Pengujian terhadap penggunaan fungsionalitas yang dimaksud berkaitan dengan kesesuaian peran fitur tombol - tombol pada aplikasi. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.2.

Tabel 5.2. Skenario Pengujian 2

Nama Skenario Pengujian	Pengujian Fungsionalitas pada Tombol di Layar Modus Kamera
Kode	SP-02
Tujuan Pengujian	Tombol <i>identify object</i> dan <i>connection</i> berjalan sesuai dengan perannya.
Kondisi Awal	Pengguna sudah menjalankan aplikasi pada perangkat bergerak.
Data Input	-
Prosedur Pengujian	1. Pengguna berada pada modus pengambilan gambar. 2. Pengguna siap untuk mencoba fitur pada aplikasi.
Hasil yang Diharapkan	Tombol <i>identify object</i> dan <i>connection</i> berjalan sesuai dengan peran fungsinya.
Hasil yang Diperoleh	Tombol <i>identify object</i> dan <i>connection</i> berhasil berjalan sesuai dengan peran fungsinya.



Gambar 5.2. Pengujian Tombol *Identify Object*



Gambar 5.3. Pengujian Tombol *Connection*

Sesuai dengan pengujian yang telah dilakukan seperti pada Gambar 5.2. dan Gambar 5.3., dapat disimpulkan bahwa penggunaan tombol pada aplikasi sudah berjalan sesuai dengan peran fungsinya. Ketika tombol *identify object* dipilih maka aplikasi akan mengambil gambar objek untuk dikirimkan ke *webservice* agar diperoleh nama dan kandungan nutrisi pada objek makanan. Tombol *connection* berjalan sesuai dengan peran fungsinya yaitu melakukan pengaturan alamat IP jaringan internet yang terhubung antara aplikasi dan *webservice*.

5.2.3. Pengujian Skenario 3 dan Evaluasi

Pada pengujian skenario 3 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada buah pisang. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.3.

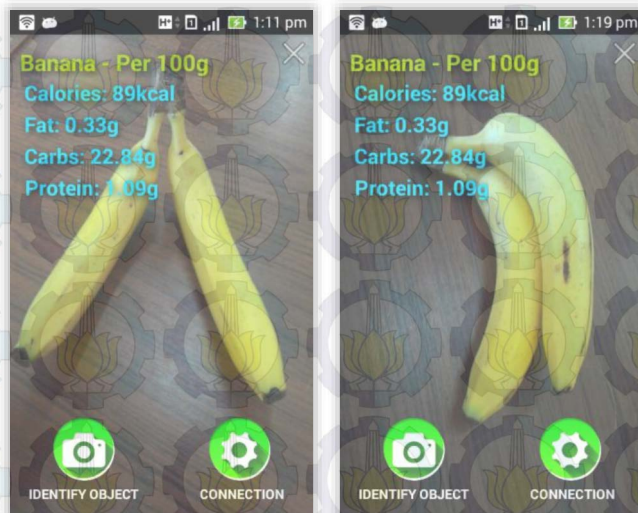
Tabel 5.3. Skenario Pengujian 3

Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Buah Pisang
Kode	SP-03
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada buah pisang.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaanodus pengambilan gambar.
Data Input	Objek buah pisang.

Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi buah pisang sebanyak 20 kali. 2. Terdapat 4 macam buah pisang dengan masing–masing buah dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi buah pisang didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi buah pisang memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi buah pisang benar sebanyak 20 dari 20 kali uji coba

Tabel 5.4. Dokumentasi Hasil Pengujian 3

Uji Coba	Hasil Identifikasi	Waktu (detik)	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	Pisang	10	11	Pisang	10
2	Pisang	11	12	Pisang	9
3	Pisang	12	13	Pisang	11
4	Pisang	11	14	Pisang	5
5	Pisang	9	15	Pisang	7
6	Pisang	10	16	Pisang	9
7	Pisang	8	17	Pisang	8
8	Pisang	11	18	Pisang	10
9	Pisang	10	19	Pisang	11
10	Pisang	9	20	Pisang	7
Tingkat kebenaran dari 20 uji coba = 100%					
Rata-rata waktu identifikasi = 9.4 detik					



Gambar 5.4. Contoh Uji Coba Pengujian Aplikasi pada Buah Pisang

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi buah pisang benar sebanyak 20 dari 20 kali uji coba. Sehingga tingkat kebenaran dari 20 kali uji coba adalah 100%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 9.4 detik. Pengujian dilakukan terhadap 4 macam buah pisang dengan modus pengambilan gambar berbeda. Contoh uji coba pengujian aplikasi pada buah pisang dapat dilihat pada Gambar 5.4.

5.2.4. Pengujian Skenario 4 dan Evaluasi

Pada pengujian skenario 4 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada buah jeruk. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama

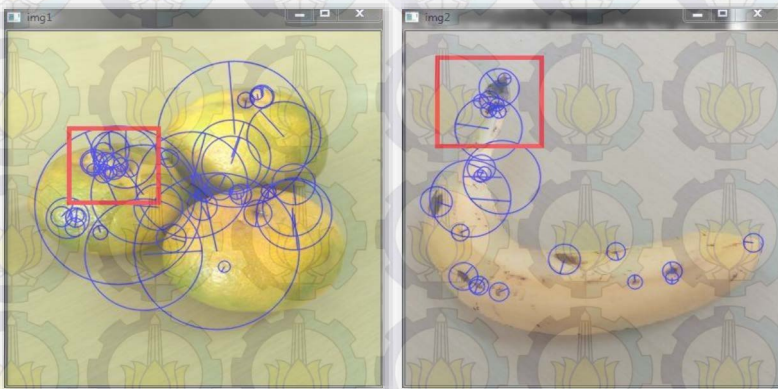
dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.5.

Tabel 5.5. Skenario Pengujian 4

Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Buah Jeruk
Kode	SP-04
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada buah jeruk.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaan modus pengambilan gambar.
Data Input	Objek buah jeruk
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi buah jeruk sebanyak 20 kali. 2. Terdapat 4 macam buah jeruk dengan masing – masing buah dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi buah jeruk didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi buah jeruk memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi buah jeruk benar sebanyak 18 dari 20 kali uji coba

Tabel 5.6. Dokumentasi Hasil Pengujian 4

Uji Coba	Hasil Identifikasi	Waktu (detik)	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	Jeruk	7	11	Jeruk	7
2	Jeruk	9	12	Jeruk	8
3	Jeruk	8	13	Jeruk	7
4	Jeruk	10	14	Jeruk	10
5	Jeruk	10	15	Jeruk	9
6	Jeruk	9	16	Pisang	11
7	Jeruk	7	17	Jeruk	8
8	Jeruk	8	18	Jeruk	9
9	Jeruk	9	19	Jeruk	10
10	Pisang	10	20	Jeruk	7
Tingkat kebenaran dari 20 uji coba = 90%					
Rata-rata waktu identifikasi = 8.65 detik					

**Gambar 5.5. Identifikasi Kemiripan Buah Jeruk dan Pisang**

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi buah jeruk benar sebanyak 18 dari 20 kali uji coba. Sedangkan 2 uji coba teridentifikasi sebagai buah pisang, hal ini dikarenakan *keypoint* pada buah jeruk memiliki

kemiripan dengan buah pisang ketika posisi pengambilan gambar untuk diidentifikasi pada uji coba 10 dan 16. Identifikasi kemiripan buah jeruk dan pisang dapat dilihat pada Gambar 5.5.



Gambar 5.6. Contoh Uji Coba Pengujian Aplikasi pada Buah Jeruk

Fitur yang diambil hanya *keypoint* saja serta bukan fitur warna dan bentuk, maka buah pisang masih terdeteksi kemiripan dengan buah jeruk. Pengujian dilakukan terhadap 4 macam buah jeruk dengan modus pengambilan gambar berbeda. Sehingga tingkat kebenaran dari 20 kali uji coba adalah 90%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 8.65 detik. Contoh uji coba pengujian aplikasi pada buah jeruk dapat dilihat pada Gambar 5.6.

5.2.5. Pengujian Skenario 5 dan Evaluasi

Pada pengujian skenario 5 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada buah

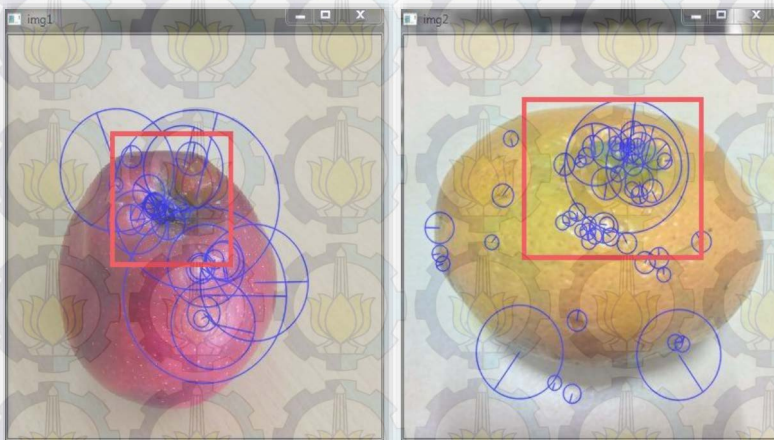
apel. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.7.

Tabel 5.7. Skenario Pengujian 5

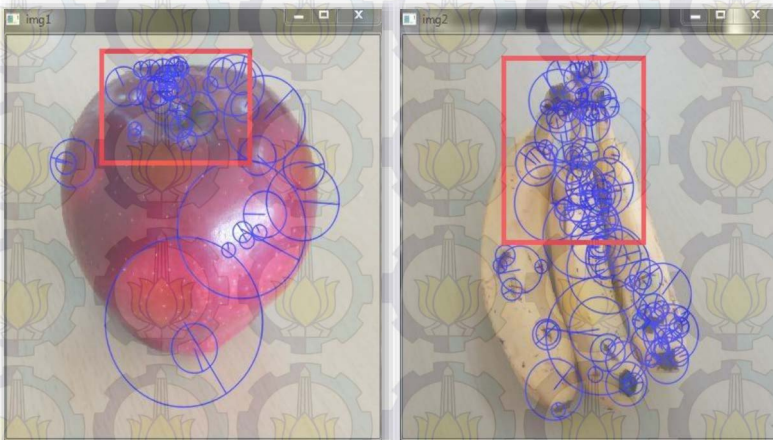
Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Buah Apel
Kode	SP-05
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada buah apel.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaan modus pengambilan gambar.
Data Input	Objek buah apel.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi buah apel sebanyak 20 kali. 2. Terdapat 4 macam buah apel dengan masing – masing buah dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi buah apel didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi buah apel memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi buah apel benar sebanyak 14 dari 20 kali uji coba

Tabel 5.8. Dokumentasi Hasil Pengujian 5

Uji Coba	Hasil Identifikasi	Waktu (detik)	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	Jeruk	11	11	Apel	9
2	Pisang	10	12	Apel	8
3	Apel	9	13	Apel	9
4	Apel	8	14	Apel	7
5	Apel	9	15	Apel	5
6	Jeruk	10	16	Apel	6
7	Jeruk	9	17	Jeruk	8
8	Apel	7	18	Jeruk	8
9	Apel	5	19	Apel	9
10	Apel	6	20	Apel	5
Tingkat kebenaran dari 20 uji coba = 70%					
Rata-rata waktu identifikasi = 7.9 detik					

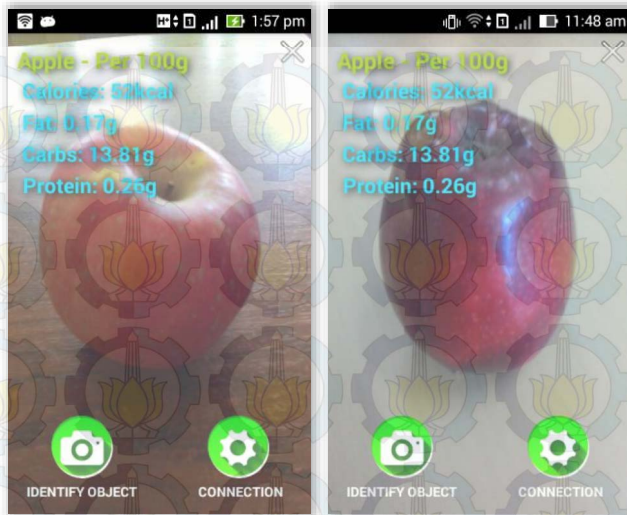
**Gambar 5.7. Identifikasi Kemiripan Buah Apel dan Jeruk**

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi buah apel benar sebanyak 14 dari 20 kali uji coba. Sedangkan 5 uji coba teridentifikasi sebagai buah jeruk dan 1 uji coba teridentifikasi sebagai buah pisang, hal ini dikarenakan *keypoint* pada buah apel memiliki kemiripan pada buah jeruk dan pisang ketika posisi pengambilan gambar untuk diidentifikasi pada uji coba 1, 2, 6, 7, 17, dan 18. Identifikasi kemiripan buah apel dan jeruk dapat dilihat pada Gambar 5.7. Sedangkan identifikasi kemiripan buah apel dan pisang dapat dilihat pada Gambar 5.8.



Gambar 5.8. Identifikasi Kemiripan Buah Apel dan Pisang

Karena fitur yang diambil hanya *keypoint* saja serta bukan fitur warna dan bentuk, maka buah jeruk dan pisang masih terdeteksi kemiripan dengan buah apel. Sehingga tingkat kebenaran dari 20 kali uji coba adalah 70%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 7.9 detik. Pengujian dilakukan terhadap 4 macam buah apel denganodus pengambilan gambar berbeda. Contoh uji coba pengujian aplikasi pada buah apel dapat dilihat pada Gambar 5.9.



Gambar 5.9. Contoh Uji Coba Pengujian Aplikasi pada Buah Apel

5.2.6. Pengujian Skenario 6 dan Evaluasi

Pada pengujian skenario 6 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Indomie. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.9.

Tabel 5.9. Skenario Pengujian 6

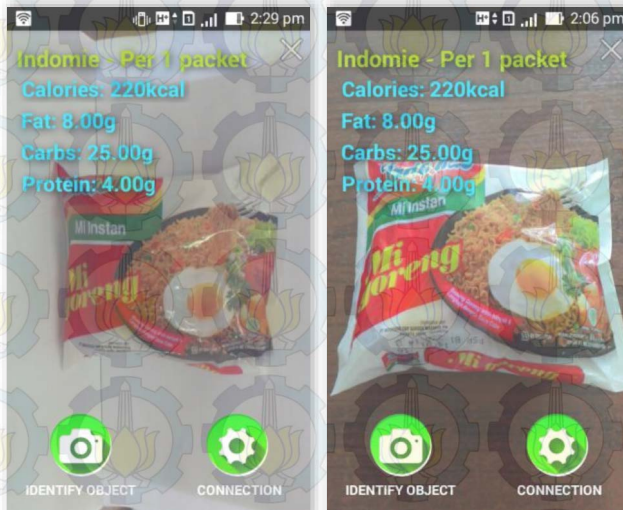
Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Produk Indomie
Kode	SP-06

Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada produk Indomie.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaan modus pengambilan gambar.
Data Input	Objek produk Indomie.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi produk Indomie sebanyak 20 kali. 2. Terdapat 4 macam posisi pengambilan gambar produk Indomie dengan masing – masing posisi dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi produk Indomie didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi produk Indomie memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi produk Indomie benar sebanyak 18 dari 20 kali uji coba

Tabel 5.10. Dokumentasi Hasil Pengujian 6

Uji Coba	Hasil Identifikasi	Waktu (detik)	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	Indomie	9	11	Indomie	9
2	Indomie	10	12	Indomie	8
3	Indomie	8	13	Indomie	10
4	Indomie	11	14	Indomie	9
5	Indomie	12	15	Indomie	11

Uji Coba	Hasil Identifikasi	Waktu (detik)	Uji Coba	Hasil Identifikasi	Waktu (detik)
6	Indomie	10	16	Indomie	9
7	Indomie	9	17	Indomie	8
8	Indomie	8	18	Indomie	10
9	Indomie	10	19	Indomie	9
10	Indomie	7	20	Indomie	8
Tingkat kebenaran dari 20 uji coba = 100%					
Rata-rata waktu identifikasi = 9.25 detik					



Gambar 5.10. Contoh Uji Coba Pengujian Aplikasi pada Produk Indomie

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi produk Indomie benar sebanyak 20 dari 20 kali uji coba. Sehingga tingkat kebenaran dari 20 kali uji coba adalah 100%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 9.25 detik. Pengujian dilakukan terhadap 4 macam posisi denganodus pengambilan gambar berbeda.

Contoh uji coba pengujian aplikasi pada produk Indomie dapat dilihat pada Gambar 5.10.

5.2.7. Pengujian Skenario 7 dan Evaluasi

Pada pengujian skenario 7 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Pringles. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.11.

Tabel 5.11. Skenario Pengujian 7

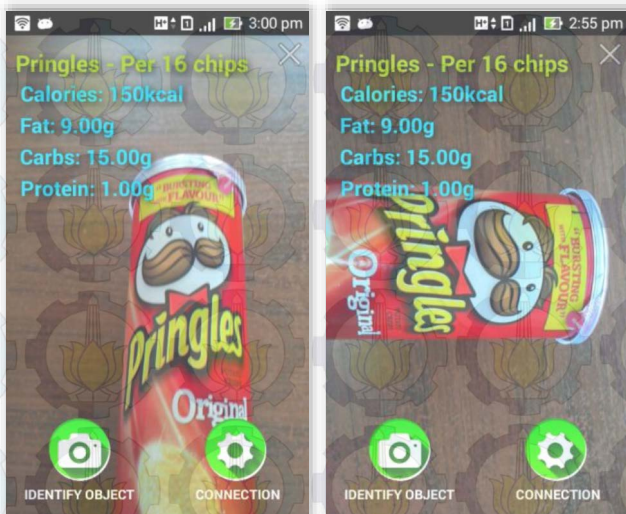
Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Produk Pringles
Kode	SP-07
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada produk Pringles.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaanodus pengambilan gambar.
Data Input	Objek produk Pringles.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi produk Pringles sebanyak 20 kali. 2. Terdapat 4 macam posisi pengambilan gambar produk Pringles dengan masing-masing posisi dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi produk Pringles

	didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi produk Pringles memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi produk Pringles benar sebanyak 20 dari 20 kali uji coba

Tabel 5.12. Dokumentasi Hasil Pengujian 7

Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	10	Pringles	11	Pringles	10
2	11	Pringles	12	Pringles	9
3	8	Pringles	13	Pringles	11
4	9	Pringles	14	Pringles	10
5	10	Pringles	15	Pringles	9
6	8	Pringles	16	Pringles	9
7	9	Pringles	17	Pringles	8
8	8	Pringles	18	Pringles	10
9	8	Pringles	19	Pringles	8
10	9	Pringles	20	Pringles	8
Tingkat kebenaran dari 20 uji coba = 100%					
Rata-rata waktu identifikasi = 9.1 detik					

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi produk Pringles benar sebanyak 20 dari 20 kali uji coba. Sehingga tingkat kebenaran dari 20 kali uji coba adalah 100%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 9.1 detik. Pengujian dilakukan terhadap 4 macam posisis dengan modus pengambilan gambar berbeda. Contoh uji coba pengujian aplikasi pada produk Pringles dapat dilihat pada Gambar 5.11.



Gambar 5.11. Contoh Uji Coba Pengujian Aplikasi pada Produk Pringles

5.2.8. Pengujian Skenario 8 dan Evaluasi

Pada pengujian skenario 8 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada buah lemon. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.13.

Tabel 5.13. Skenario Pengujian 8

Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Buah Lemon
-------------------------	---

Kode	SP-08
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada buah lemon.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaan modus pengambilan gambar.
Data Input	Objek buah lemon.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi buah lemon sebanyak 20 kali. 2. Terdapat 4 macam produk Pringles dengan masing-masing buah dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi buah lemon didokumentasikan kebenaran dan kesalahannya
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi buah lemon memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi buah lemon benar sebanyak 11 dari 20 kali uji coba

Tabel 5.14. Dokumentasi Hasil Pengujian 8

Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	7	Lemon	11	Lemon	8
2	6	Apel	12	Pisang	7
3	9	Pisang	13	Pisang	9
4	6	Lemon	14	Lemon	7
5	7	Pisang	15	Lemon	8
6	8	Lemon	16	Lemon	9
7	8	Lemon	17	Lemon	9

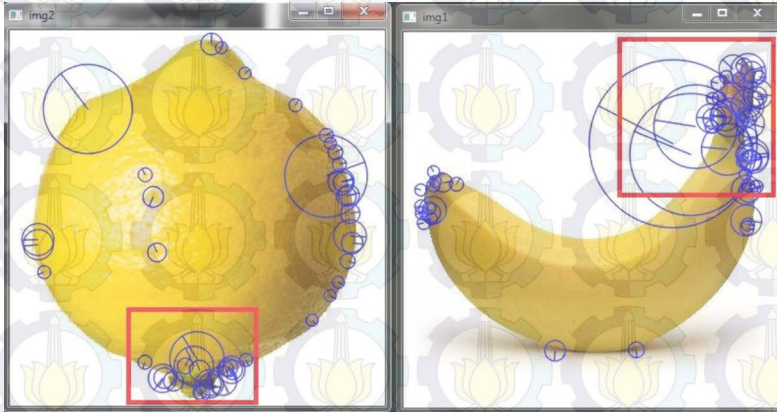
Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
8	6	Apel	18	Pisang	9
9	7	Apel	19	Lemon	8
10	8	Pisang	20	Lemon	9
Tingkat kebenaran dari 20 uji coba = 55%					
Rata-rata waktu identifikasi = 7.75 detik					



Gambar 5.12. Contoh Uji Coba Pengujian Aplikasi pada Buah Lemon

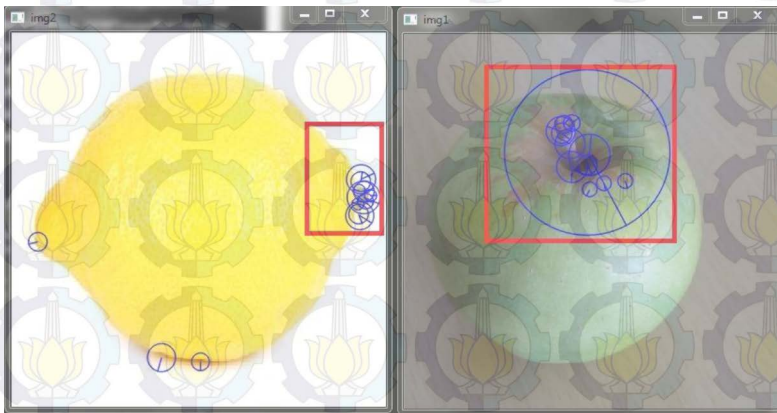
Gambar 5.12. adalah contoh uji coba pengujian aplikasi pada buah lemon. Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi buah lemon benar sebanyak 11 dari 20 kali uji coba. Sedangkan 3 uji coba teridentifikasi sebagai buah apel dan 6 uji coba teridentifikasi sebagai buah pisang, hal ini dikarenakan *keypoint* pada buah lemon memiliki kemiripan pada buah apel dan pisang. Faktor lain yang mempengaruhi adalah posisi modus kamera pada saat pengambilan gambar. Sehingga

diperoleh rata-rata prosentase kebenaran dalam mengidentifikasi objek sebesar 55%. Rata-rata waktu yang dibutuhkan untuk identifikasi selama 7.75 detik. Identifikasi kemiripan buah lemon dan pisang dapat dilihat pada Gambar 5.13.



Gambar 5.13. Identifikasi Kemiripan Buah Lemon dan Pisang

Sedangkan identifikasi kemiripan buah lemon dan apel dapat dilihat pada Gambar 5.14.



Gambar 5.14. Identifikasi Kemiripan Buah Lemon dan Apel

5.2.9. Pengujian Skenario 9 dan Evaluasi

Pada pengujian skenario 9 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Pocky. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.15.

Tabel 5.15. Skenario Pengujian 9

Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Produk Pocky
Kode	SP-09
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada produk Pocky.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaan modus pengambilan gambar.
Data Input	Objek produk Pocky.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi produk Pocky sebanyak 20 kali. 2. Terdapat 4 macam posisi pengambilan gambar produk Pocky dengan masing-masing posisi dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi produk Pocky didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.

Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi produk Pocky memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi produk Pocky benar sebanyak 20 dari 20 kali uji coba

Tabel 5.16. Dokumentasi Hasil Pengujian 9

Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	9	Pocky	11	Pocky	9
2	10	Pocky	12	Pocky	10
3	11	Pocky	13	Pocky	10
4	9	Pocky	14	Pocky	9
5	9	Pocky	15	Pocky	10
6	10	Pocky	16	Pocky	11
7	9	Pocky	17	Pocky	11
8	9	Pocky	18	Pocky	10
9	10	Pocky	19	Pocky	9
10	11	Pocky	20	Pocky	10
Tingkat kebenaran dari 20 uji coba = 100 %					
Rata-rata waktu identifikasi = 9.8 detik					

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi produk Pocky benar sebanyak 20 dari 20 kali uji coba. Sehingga tingkat kebenaran dari 20 kali uji coba adalah 100%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 9.8 detik. Pengujian dilakukan terhadap 4 macam posisi dengan modus pengambilan gambar berbeda. Contoh uji coba pengujian aplikasi pada produk Pocky dapat dilihat pada Gambar 5.15.



Gambar 5.15. Contoh Uji Coba Pengujian Aplikasi pada Produk Pocky

5.2.10. Pengujian Skenario 10 dan Evaluasi

Pada pengujian skenario 10 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Oreo. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.17.

Tabel 5.17. Skenario Pengujian 10

Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Produk Oreo
-------------------------	--

Kode	SP-10
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada produk Oreo.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaan modus pengambilan gambar.
Data Input	Objek produk Oreo.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi produk Oreo sebanyak 20 kali. 2. Terdapat 4 macam posisi pengambilan gambar produk Oreo dengan masing-masing posisi dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi produk Oreo didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi produk Oreo memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi produk Oreo benar sebanyak 20 dari 20 kali uji coba

Tabel 5.18. Dokumentasi Hasil Pengujian 10

Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	8	Oreo	11	Oreo	10
2	7	Oreo	12	Oreo	9
3	7	Oreo	13	Oreo	8
4	9	Oreo	14	Oreo	10

Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
5	9	Oreo	15	Oreo	9
6	8	Oreo	16	Oreo	9
7	7	Oreo	17	Oreo	8
8	9	Oreo	18	Oreo	11
9	9	Oreo	19	Oreo	10
10	11	Oreo	20	Oreo	10
Tingkat kebenaran dari 20 uji coba = 100 %					
Rata-rata waktu identifikasi = 8.9 detik					



Gambar 5.16. Contoh Uji Coba Pengujian Aplikasi pada Produk Oreo

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi produk Oreo benar sebanyak 20 dari 20 kali uji coba. Sehingga tingkat kebenaran dari 20 kali uji coba adalah 100%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 8.9 detik. Pengujian dilakukan terhadap

4 macam posisi denganodus pengambilan gambar berbeda. Contoh uji coba pengujian aplikasi pada produk Oreo dapat dilihat pada Gambar 5.16.

5.2.11. Pengujian Skenario 11 dan Evaluasi

Pada pengujian skenario 11 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Snickers. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.19.

Tabel 5.19. Skenario Pengujian 11

Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan Kandungan Nutrisi pada Produk Snickers
Kode	SP-11
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada produk Snickers.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaanodus pengambilan gambar.
Data Input	Objek produk Snickers.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi produk Snickers sebanyak 20 kali. 2. Terdapat 4 macam posisi pengambilan gambar produk Snickers dengan masing-masing posisi dilakukan 5 kali uji coba.

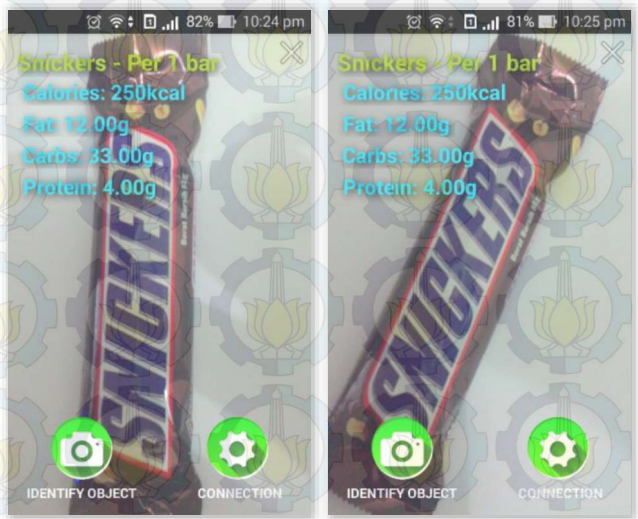
	3. Hasil uji coba identifikasi nama dan nutrisi produk Snickers didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi produk Snickers memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi produk Snickers benar sebanyak 20 dari 20 kali uji coba

Tabel 5.20. Dokumentasi Hasil Pengujian 11

Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	11	Snickers	11	Snickers	9
2	10	Snickers	12	Snickers	10
3	12	Snickers	13	Snickers	12
4	11	Snickers	14	Snickers	12
5	11	Snickers	15	Snickers	10
6	12	Snickers	16	Snickers	11
7	10	Snickers	17	Snickers	10
8	13	Snickers	18	Snickers	10
9	10	Snickers	19	Snickers	10
10	12	Snickers	20	Snickers	11
Tingkat kebenaran dari 20 uji coba = 100 %					
Rata-rata waktu identifikasi = 10.85 detik					

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi produk Snickers benar sebanyak 20 dari 20 kali uji coba. Sehingga tingkat kebenaran dari 20 kali uji coba adalah 100%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 10.85 detik. Pengujian dilakukan terhadap 4 macam posisi dengan modus pengambilan gambar

berbeda. Contoh uji coba pengujian aplikasi pada produk Snickers dapat dilihat pada Gambar 5.17.



Gambar 5.17. Contoh Uji Coba Pengujian Aplikasi pada Produk Snickers

5.2.12. Pengujian Skenario 12 dan Evaluasi

Pada pengujian skenario 12 ini dilakukan pengujian terhadap identifikasi objek berupa nama dan kandungan nutrisi pada produk Tango. Dalam pengujian, dilakukan 20 kali uji coba. Pengujian juga bertujuan untuk mengetahui tingkat kebenaran identifikasi nama dan kandungan nutrisi. Setiap kali uji coba dilakukan dokumentasi kebenaran dan kesalahannya. Skenario dan hasil pengujian dapat dilihat pada Tabel 5.21.

Tabel 5.21. Skenario Pengujian 12

Nama Skenario Pengujian	Pengujian Terhadap Identifikasi Objek Berupa Nama dan
-------------------------	---

	Kandungan Nutrisi pada Produk Tango
Kode	SP-12
Tujuan Pengujian	Mengidentifikasi kebenaran nama objek dan kandungan nutrisi pada produk Tango.
Kondisi Awal	Pengguna menggunakan aplikasi dalam keadaan modus pengambilan gambar.
Data Input	Objek produk Tango.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Dilakukan uji coba identifikasi nama dan nutrisi produk Tango sebanyak 20 kali. 2. Terdapat 4 macam posisi pengambilan gambar produk Tango dengan masing-masing posisi dilakukan 5 kali uji coba. 3. Hasil uji coba identifikasi nama dan nutrisi produk Tango didokumentasikan kebenaran dan kesalahannya serta waktu yang diperlukan.
Hasil yang Diharapkan	Hasil uji coba identifikasi nama dan nutrisi produk Tango memiliki tingkat kebenaran 20 dari 20 kali uji coba.
Hasil yang Diperoleh	Hasil uji coba identifikasi nama dan nutrisi produk Tango benar sebanyak 20 dari 20 kali uji coba

Tabel 5.22. Dokumentasi Hasil Pengujian 12

Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
1	11	Tango	11	Tango	11
2	13	Tango	12	Tango	11

Uji Coba	Waktu (detik)	Hasil Identifikasi	Uji Coba	Hasil Identifikasi	Waktu (detik)
3	10	Tango	13	Tango	10
4	10	Tango	14	Tango	10
5	13	Tango	15	Tango	11
6	12	Tango	16	Tango	12
7	11	Tango	17	Tango	13
8	13	Tango	18	Tango	10
9	11	Tango	19	Tango	11
10	12	Tango	20	Tango	12
Tingkat kebenaran dari 20 uji coba = 100 %					
Rata-rata waktu identifikasi = 11.35 detik					



Gambar 5.18. Contoh Uji Coba Pengujian Aplikasi pada Produk Tango

Dari pengujian yang dilakukan, diperoleh hasil uji coba identifikasi nama dan nutrisi produk Tango benar sebanyak 20 dari 20 kali uji coba. Sehingga tingkat kebenaran dari 20 kali uji coba

adalah 100%. Rata-rata waktu untuk identifikasi nama dan kandungan nutrisi selama 11.35 detik. Pengujian dilakukan terhadap 4 macam posisi dengan modus pengambilan gambar berbeda. Contoh uji coba pengujian aplikasi pada produk Snickers dapat dilihat pada Gambar 5.18.

5.2.13. Pengujian Skenario 13 dan Evaluasi

Pengujian skenario 13 ini merupakan kalkulasi kebenaran aplikasi dalam mengidentifikasi nama dan nutrisi. Kalkulasi kebenaran aplikasi adalah hasil dari keseluruhan uji coba dari pengujian 3 hingga 12. Hasil pengujian dapat dilihat pada Tabel 5.23.

Tabel 5.23. Hasil Pengujian 13

		Target Kelas									
		Pisang	Jeruk	Apel	Indomie	Pringles	Lemon	Pocky	Oreo	Snickers	Tango
Hasil Uji Coba	Pisang	20	2	1			6				
	Jeruk		18	5							
	Apel			14			3				
	Indomie				20						
	Pringles					20					
	Lemon						11				
	Pocky							20			
	Oreo								20		
	Snickers									20	
	Tango										20

Sedangkan hasil prosentase kebenaran identifikasi objek makanan atau produk makanan dapat dilihat pada Tabel 5.24. Hasil kalkulasi prosentase kebenaran merupakan rata-rata dari pengujian skenario 3 hingga pengujian skenario 12.

Tabel 5.24. Hasil Kalkulasi Prosentase Kebenaran

Nama Objek	Jumlah Benar	Prosentase Benar
Pisang	20	100%
Jeruk	18	90%
Apel	14	70%
Indomie	20	100%
Pringles	20	100%
Lemon	11	55%
Pocky	20	100%
Oreo	20	100%
Snickers	20	100%
Tango	20	100%
RATA – RATA		92%

Tabel 5.25. Hasil Kalkulasi Waktu Identifikasi

Nama Objek	Rata-Rata Waktu Identifikasi (detik)
Pisang	9.4
Jeruk	8.65
Apel	7.9
Indomie	9.25
Pringles	9.1
Lemon	7.75
Pocky	9.8
Oreo	8.9
Snickers	10.85
Tango	11.35
RATA–RATA KESELURUHAN	9.295

Pada pengujian skenario 3 hingga 12 didapatkan rata-rata prosentase kebenaran aplikasi dalam mengidentifikasi objek makanan atau produk makanan sebesar 92%. Identifikasi kebenaran pada buah apel dan lemon memiliki prosentase yang rendah dibandingkan dengan 8 objek yang lainnya. Hal ini dikarenakan fitur yang diambil hanya *keypoint* saja serta bukan

fitur warna dan bentuk. Faktor lain yang mempengaruhi yaitu modus kamera pada saat posisi pengambilan gambar untuk diidentifikasi. Sedangkan lamanya waktu yang diperlukan untuk mengidentifikasi nama dan kandungan nutrisi tergantung pada kompleksitas fitur *keypoint* pada gambar. Hasil kalkulasi waktu identifikasi dapat dilihat pada Tabel 5.25.

5.2.14. Pengujian Skenario 14 dan Evaluasi

Pada pengujian skenario 14 ini dilakukan terhadap pengguna secara langsung yaitu dengan melibatkan Mahasiswa dan kalangan masyarakat sekitar ITS sebagai pengguna langsung aplikasi. Dalam pengujian, dilakukan penyebaran kuesioner sebagai timbal balik dari pengguna ke pengembang aplikasi.

5.2.14.1. Daftar Responden Sebagai Penguji

Pada subbab ini ditunjukkan daftar Responden yang bertindak sebagai penguji coba aplikasi yang telah dibangun. Daftar Responden sebagai penguji dapat dilihat pada Tabel 5.26.

Tabel 5.26. Daftar Responden Sebagai Penguji

Nama	Rentang Umur (tahun)	Jenis Kelamin
Responden 1	21 – 30	Laki-laki
Responden 2	21 – 30	Laki-laki
Responden 3	21 – 30	Perempuan
Responden 4	21 – 30	Laki-laki
Responden 5	21 – 30	Perempuan
Responden 6	21 – 30	Perempuan
Responden 7	21 – 30	Perempuan
Responden 8	31 – 40	Perempuan
Responden 9	13 – 20	Perempuan
Responden 10	21 – 30	Laki-laki

5.2.14.2. Hasil Uji Coba Kepada Responden Sebagai Penguji

Uji coba kepada pengguna sebagai penguji menggunakan metode penyebaran kuesioner sebagai timbal balik dari pengguna ke pengembang setelah menggunakan aplikasi. Hasil uji coba kepada pengguna sebagai penguji dapat dilihat pada Tabel 5.27.

Tabel 5.27. Hasil Uji Coba Kepada Responden Sebagai Penguji

Kode	Indikator	Rata – Rata Nilai (Rentang 1-5)
UC-1	Pengguna selalu memperhatikan nutrisi makanan yang dikonsumsi sehari-hari	2.4
UC-2	Kemudahan aplikasi ketika digunakan untuk mendapatkan informasi nutrisi	3.7
UC-3	Ketertarikan pengguna terhadap aplikasi untuk mengontrol konsumsi nutrisi sehari-hari	3.9
UC-4	Kebenaran aplikasi dalam mendeteksi nama dan nutrisi objek makanan	3.3
UC-5	Kemudahan menavigasikan icon aplikasi dalam penggunaan aplikasi	4.2
RATA – RATA TOTAL		3.5

Sedangkan detail pengujian pada setiap aspek indikator kuesioner yang disebar kepada pengguna sebagai penguji langsung dapat dilihat pada Tabel 5.28.

Tabel 5.28. Detil Pengujian pada Setiap Aspek Indikator

Nama Pengguna	Aspek Indikator				
	UC-1	UC-2	UC-3	UC-4	UC-5
Responden 1	1	4	4	4	4
Responden 2	3	4	4	4	4

Nama Pengguna	Aspek Indikator				
	UC-1	UC-2	UC-3	UC-4	UC-5
Responden 3	3	3	3	2	5
Responden 4	2	5	5	4	5
Responden 5	4	4	4	3	4
Responden 6	4	3	3	3	4
Responden 7	4	4	4	4	5
Responden 8	1	3	4	3	3
Responden 9	1	3	4	3	4
Responden 10	2	4	4	3	4
TOTAL	22	33	35	30	38

BAB VI

KESIMPULAN DAN SARAN

Pada bab ini diberikan kesimpulan yang diambil selama pengerjaan tugas akhir serta saran-saran tentang pengembangan yang dapat dilakukan terhadap tugas akhir ini di masa yang akan datang.

6.1. Kesimpulan

Dari proses pengerjaan selama perancangan, implementasi, dan proses pengujian aplikasi yang dilakukan, dapat diambil kesimpulan sebagai berikut.

1. Aplikasi dapat mengidentifikasi jenis makanan atau produk makanan dari masukan berupa gambar menggunakan metode klasifikasi gambar, sehingga bisa didapatkan informasi berupa nama dari objek makanan atau produk makanan yang diidentifikasi.
2. Metode yang digunakan untuk ekstraksi fitur *keypoint* gambar adalah SURF (*Speeded-Up Robust Feature*) yang sudah disediakan oleh pustaka OpenCV. Setelah mendapatkan fitur *keypoint*, dilakukan perhitungan jarak terdekat menggunakan *Euclidian Distance* untuk klasifikasi objek gambar. Apabila jarak yang dihitung semakin kecil maka kemiripan semakin besar.
3. Aplikasi dapat mengidentifikasi kandungan nutrisi berupa kalori, karbohidrat, protein dan lemak di dalam makanan atau produk makanan menggunakan basis data pada *webservice Fatsecret*. Masukan untuk *webservice Fatsecret* adalah hasil klasifikasi gambar di *webservice* yaitu berupa nama makanan atau produk makanan. Nama makanan atau produk makanan kemudian dikirimkan ke pencarian kandungan nutrisi pada basis data *Fatsecret*. Keluaran

pencarian kandungan nutrisi pada basis data *Fatsecret* berupa rangkuman nutrisi per takaran saji.

4. Pengujian kebenaran aplikasi dalam mengidentifikasi nama dan informasi nutrisi yang terkandung memiliki rata-rata prosentase kebenaran sebesar 92% dengan rata-rata waktu 9.295 detik.
5. Identifikasi buah apel dan lemon memiliki prosentase akurasi rendah dibandingkan dengan objek yang lainnya. Hal ini dikarenakan fitur yang diambil hanya *keypoint* saja, serta bukan fitur warna dan bentuk. Faktor lain yang mempengaruhi yaitu posisi modus kamera pada saat pengambilan gambar objek untuk diidentifikasi.
6. Aplikasi dapat menampilkan informasi nutrisi berupa kalori, karbohidrat, protein dan lemak per takaran saji menggunakan teknologi *augmented reality* teks berbasis *android* sehingga lebih menarik.

6.2. Saran

Berikut saran-saran untuk pengembangan dan perbaikan sistem di masa yang akan datang. Di antaranya adalah sebagai berikut:

1. Untuk memberikan kesan yang lebih informatif perlu ditampilkan suatu grafik yang dinamis berdasarkan masukan informasi kandungan nutrisi dari *Fatsecret* API. Sampai saat ini pengembang belum menemukan cara bagaimana menampilkan grafik secara dinamis pada permukaan layar modus kamera *android*.
2. Diperlukan suatu SDK (*Software Development Kit*) khusus untuk menampilkan *augmented reality* teks tanpa menggunakan proses *rendering* dari teks menjadi gambar sehingga bisa dinamis. Untuk saat ini pengembang menampilkan informasi nutrisi dengan membangun secara tradisional pada layar permukaan modus kamera *android*.

DAFTAR PUSTAKA

- [1] World Health Organization, "Obesity and Overweight Fact Sheet," World Health Organization, 2nd March 2013. [Online]. Available: <http://www.who.int/mediacentre/factsheets/fs311/en/>. [Accessed 6th November 2014].
- [2] S. Langley-Evans, "Human Nutrition and Dietetics," *Journal of Human Nutrition and Dietetics*, vol. 79, p. 48, Juni 2013.
- [3] L. A. Times, "Diet," Los Angeles Times, 21 October 2014. [Online]. Available: <http://articles.latimes.com/keyword/diet>. [Accessed 7th November 2014].
- [4] Yuku, "Nutrition Facts Apps Store," Nutrition Facts, 30th September 2014. [Online]. Available: <https://play.google.com/store/apps/details?id=yuku.usdanu.t.lite&hl=en>. [Accessed 7th November 2014].
- [5] H. A. N. M. A. Muhammad Zulfikar Bayu, "Nutritional Information Visualization Using Mobile Augmented Reality Technology," *ScienceDirect*, vol. 11, no. The 4th International Conference on Electrical Engineering and Informatics (ICEEI 2013), pp. 396-402, 2013.
- [6] "Nutrition Definition," 2014. [Online]. Available: <http://www.who.int/topics/nutrition/en/>. [Accessed 14th November 2014].
- [7] "Calorie Counting - A Guide to Calories & Weight Control," 2014. [Online]. Available: <http://www.acaloriecounter.com/calorie-counting.php>. [Accessed 14th November 2014].

- [8] A. W. Mahastama, "Pemanfaatan Computer Vision: Augmented Reality," 2014. [Online]. Available: http://lecturer.ukdw.ac.id/~mahas/dossier/comvis_08.pdf. [Accessed 14th November 2014].
- [9] "OS Android," DigitalGlobe Astrium, 2014. [Online]. Available: <http://www.android.com/history/>. [Accessed 14th November 2014].
- [10] developer.android.com, "Android Studio," developer.android.com, October 2014. [Online]. Available: <http://developer.android.com/tools/studio/index.html>. [Accessed 14th November 2014].
- [11] H. Bay, "Speeded-Up Robust Features (SURF)," *ScienceDirect*, vol. 110, no. 13, p. 346–359, 2008.
- [12] OpencvDevTeam, "OpenCV 2.4.11.0 documentation," 25th February 2014. [Online]. Available: <http://docs.opencv.org/modules/core/doc/intro.html>. [Accessed 14th November 2014].
- [13] T. S. Waqqas Sheikh, "Prototype of an Image Recognition Based Calorie," *Heriot Watt University*, vol. 129, no. Heriot Watt University, p. 129, 2013.
- [14] "Introduction to Web Services," w3schools, 28th October 2014. [Online]. Available: http://www.w3schools.com/webservices/ws_intro.asp. [Accessed 15th November 2014].
- [15] A. Ronacher, "flask.pocoo.org," 2013. [Online]. Available: <http://flask.pocoo.org/docs/0.10/foreword/#what-does-micro-mean>. [Accessed 16th May 2015].

BAB VI

KESIMPULAN DAN SARAN

Pada bab ini diberikan kesimpulan yang diambil selama pengerjaan tugas akhir serta saran-saran tentang pengembangan yang dapat dilakukan terhadap tugas akhir ini di masa yang akan datang.

6.1. Kesimpulan

Dari proses pengerjaan selama perancangan, implementasi, dan proses pengujian aplikasi yang dilakukan, dapat diambil kesimpulan sebagai berikut.

1. Aplikasi dapat mengidentifikasi jenis makanan atau produk makanan dari masukan berupa gambar menggunakan metode klasifikasi gambar, sehingga bisa didapatkan informasi berupa nama dari objek makanan atau produk makanan yang diidentifikasi.
2. Metode yang digunakan untuk ekstraksi fitur *keypoint* gambar adalah SURF (*Speeded-Up Robust Feature*) yang sudah disediakan oleh pustaka OpenCV. Setelah mendapatkan fitur *keypoint*, dilakukan perhitungan jarak terdekat menggunakan *Euclidian Distance* untuk klasifikasi objek gambar. Apabila jarak yang dihitung semakin kecil maka kemiripan semakin besar.
3. Aplikasi dapat mengidentifikasi kandungan nutrisi berupa kalori, karbohidrat, protein dan lemak di dalam makanan atau produk makanan menggunakan basis data pada *webservice Fatsecret*. Masukan untuk *webservice Fatsecret* adalah hasil klasifikasi gambar di *webservice* yaitu berupa nama makanan atau produk makanan. Nama makanan atau produk makanan kemudian dikirimkan ke pencarian kandungan nutrisi pada basis data *Fatsecret*. Keluaran

pencarian kandungan nutrisi pada basis data *Fatsecret* berupa rangkuman nutrisi per takaran saji.

4. Pengujian kebenaran aplikasi dalam mengidentifikasi nama dan informasi nutrisi yang terkandung memiliki rata-rata prosentase kebenaran sebesar 92% dengan rata-rata waktu 9.295 detik.
5. Identifikasi buah apel dan lemon memiliki prosentase akurasi rendah dibandingkan dengan objek yang lainnya. Hal ini dikarenakan fitur yang diambil hanya *keypoint* saja, serta bukan fitur warna dan bentuk. Faktor lain yang mempengaruhi yaitu posisi modus kamera pada saat pengambilan gambar objek untuk diidentifikasi.
6. Aplikasi dapat menampilkan informasi nutrisi berupa kalori, karbohidrat, protein dan lemak per takaran saji menggunakan teknologi *augmented reality* teks berbasis *android* sehingga lebih menarik.

6.2. Saran

Berikut saran-saran untuk pengembangan dan perbaikan sistem di masa yang akan datang. Di antaranya adalah sebagai berikut:

1. Untuk memberikan kesan yang lebih informatif perlu ditampilkan suatu grafik yang dinamis berdasarkan masukan informasi kandungan nutrisi dari *Fatsecret* API. Sampai saat ini pengembang belum menemukan cara bagaimana menampilkan grafik secara dinamis pada permukaan layar modus kamera *android*.
2. Diperlukan suatu SDK (*Software Development Kit*) khusus untuk menampilkan *augmented reality* teks tanpa menggunakan proses *rendering* dari teks menjadi gambar sehingga bisa dinamis. Untuk saat ini pengembang menampilkan informasi nutrisi dengan membangun secara tradisional pada layar permukaan modus kamera *android*.

LAMPIRAN (A) KODE SUMBER

```
public class MainActivity extends ActionBarActivity
implements View.OnClickListener {

    private Camera mCamera = null;
    private CameraView mCameraView = null;
    private String filepath;
    private TextView objects, calories, fat, carbs,
    protein;
    SharedPreferences sp;

    private Camera.PictureCallback mPicture = new
    Camera.PictureCallback() {

        @Override
        public void onPictureTaken(byte[] data,
        Camera camera) {

            File pictureFile = new
            File(Environment.getExternalStorageDirectory(),
            "testTA.jpg");

            filepath =
            pictureFile.getAbsolutePath();
            try {
                FileOutputStream fos = new
                FileOutputStream(pictureFile);
                fos.write(data);
                fos.close();
            } catch (Exception e) {
                e.printStackTrace();
            }
            new UploadImage(filepath).execute();
        }
    };

    @Override
    protected void onCreate(Bundle
    savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        sp =
        PreferenceManager.getDefaultSharedPreferences(this);
        PreferenceManager.setDefaultValues(this,
```

```

R.xml.preference_screen, false);

        objects = (TextView)
findViewById(R.id.objectTxt);
        calories = (TextView)
findViewById(R.id.caloriesTxt);
        fat = (TextView) findViewById(R.id.fatTxt);
        carbs = (TextView)
findViewById(R.id.carbsTxt);
        protein = (TextView)
findViewById(R.id.proteinTxt);

        ImageView imgPic = (ImageView)
findViewById(R.id.btn_identify);
        imgPic.setOnClickListener(this);

        ImageView imgSet = (ImageView)
findViewById(R.id.btn_setting);
        imgSet.setOnClickListener(this);

        ImageButton imgClose =
(ImageButton) findViewById(R.id.imgClose);
        imgClose.setOnClickListener(this);
    }

```

Kode Sumber 7.1. Kode MainActivity.java (Bagian 1)

```

@Override
protected void onPause() {
    super.onPause();
    try{
        mCamera.stopPreview();

        mCameraView.getHolder().removeCallback(mCameraView);
        mCamera.release();
        mCamera = null;
    }catch (Exception e){
        e.printStackTrace();
    }
}

@Override
protected void onResume() {
    super.onResume();
    getCameraInstance();
}

```

```

}
@Override
public void onClick(View v) {
    if (v.getId() == R.id.btn_identify) {
        mCamera.takePicture(null, null, mPicture);
    } else if (v.getId() == R.id.btn_setting) {
        Intent intent = new Intent(this,
        SettingActivity.class);
        this.startActivity(intent);
    } else if (v.getId() == R.id.imgClose) {
        System.exit(1);
    }
}
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu_main,
    menu);
    return true;
}

private void getCameraInstance() {
    try {
        mCamera = Camera.open();
    } catch (Exception e) {
        Log.d("ERROR", "Failed to get camera: " +
        e.getMessage());
    }
    if (mCamera != null) {
        Camera.Parameters params =
        mCamera.getParameters();
        params.setFocusMode(Camera.Parameters.FOCUS_MODE_AUT
        O);

        params.setPictureSize(400, 400);
        params.setRotation(90);
        mCamera.setParameters(params);
        mCameraView = new CameraView(this, mCamera);
        FrameLayout camera_view =
        (FrameLayout) findViewById(R.id.camera_view);
        camera_view.addView(mCameraView);
    }
}
}

```

Kode Sumbur 7.2. Kode *MainActivity.java* (Bagian 2)


```

private void updateInfo(String hasil){
    String[] split0 = hasil.split("#");
    String[] split1 = split0[1].split("-");
    String[] split2 = split1[1].split("\\|");
    objects.setText(split0[0] + " - " + split1[0]);
    calories.setText(split2[0]);
    fat.setText(split2[1]);
    carbs.setText(split2[2]);
    protein.setText(split2[3]);
    Handler h = new Handler();
    h.postDelayed(new Runnable() {
        public void run() {
            //change your text here
            objects.setText("");
            calories.setText("");
            fat.setText("");
            carbs.setText("");
            protein.setText("");
        }
    }, 10000);
}
@Override
public boolean onOptionsItemSelected(MenuItem item)
{
    int id = item.getItemId();
    if (id == R.id.action_settings) {
        Intent intent = new Intent(this,
        SettingActivity.class);
        this.startActivity(intent);
        return true;
    }
    return super.onOptionsItemSelected(item);
}

```

Kode Sumber 7.3. Kode MainActivity.java (Bagian 3)

```

@Override
protected String doInBackground(String... params) {
    String urlAddr =
    sp.getString(SettingActivity.KEY_PREF_SERVER_ADDRESS
    , "");
    File file = new File(namaImage);
    HttpURLConnection httpConn = null;
    DataOutputStream dos = null;

```

```

String lineEnd = "\r\n";
String twoHyphens = "--";
String boundary = "*****";
int bytesRead, bytesAvailable, maxBufferSize;
byte[] buffer;

try {
    FileInputStream fus = new
FileInputStream(file);
    URL url = new URL(urlAddr);
    httpConn = (URLConnection)
url.openConnection();
    httpConn.setDoInput(true);
    httpConn.setDoOutput(true);
    httpConn.setUseCaches(false);
    httpConn.setRequestMethod("POST");
    httpConn.setRequestProperty("Connection",
"Keep-Alive");
    httpConn.setRequestProperty("ENCTYPE",
"multipart/form-data");
    httpConn.setRequestProperty("Content-Type",
"multipart/form-data;boundary=" + boundary);

    dos = new
DataOutputStream(httpConn.getOutputStream());
    dos.writeBytes(twoHyphens + boundary +
lineEnd);
    dos.writeBytes("Content-Disposition: form-
data; name=\"file\"; filename=\"" + "testTA.jpg\" +
lineEnd);
    dos.writeBytes(lineEnd);

    bytesAvailable = fus.available();
    maxBufferSize = 1024 * 1024;
    buffer = new byte[bytesAvailable];
    bytesRead = fus.read(buffer, 0,
bytesAvailable);
    while (bytesRead > 0) {
        dos.write(buffer, 0, bytesAvailable);
        bytesAvailable = fus.available();
        bytesAvailable =
Math.min(bytesAvailable, maxBufferSize);
        bytesRead = fus.read(buffer, 0,

```

```

bytesAvailable);
    }
    dos.writeBytes(lineEnd);
    dos.writeBytes(twoHyphens + boundary +
twoHyphens + lineEnd);

    fus.close();
    dos.flush();
    dos.close();

    int serverResponseCode =
httpConn.getResponseCode();
    String serverResponseMessage =
httpConn.getResponseMessage();
    InputStream is = httpConn.getInputStream();
    final String hasil =
convertStreamToString(is);
    Log.i("hasil = ", hasil);

    runOnUiThread(new Runnable() {
        @Override
        public void run() {
            updateInfo(hasil);
        }
    });

    Log.i("uploadFile", "HTTP Response is : " +
serverResponseMessage + ": " + serverResponseCode);
} catch (Exception e) {
    e.printStackTrace();
}
return null;
}
@Override
protected void onProgressUpdate(String...
values) {
    super.onProgressUpdate(values);
    if (values[0].matches("0")){
        Toast.makeText(getApplicationContext(),
"Transfer image failed!!",
Toast.LENGTH_LONG).show();
    }else{
        Toast.makeText(getApplicationContext(),
"Transfer image is completed!!",

```

```

Toast.LENGTH_LONG).show();
    }
}

@Override
protected void onPostExecute(String s) {
    super.onPostExecute(s);
    pDialog.dismiss();
    mCamera.startPreview();
}
}

```

Kode Sumber 7.4. Kode MainActivity.java (Bagian 4)

```

public static int getExifOrientation(String
filepath){
    int degree = 0;
    ExifInterface exif = null;

    try {
        exif = new ExifInterface(filepath);
    } catch (Exception ex){
        ex.printStackTrace();
    }
    if (exif != null) {
        int orientation =
exif.getAttributeInt(ExifInterface.TAG_ORIENTATION,
-1);
        if (orientation != -1) {
            switch (orientation) {
                case
ExifInterface.ORIENTATION_ROTATE_90:
                    degree = 90;
                    break;
                case
ExifInterface.ORIENTATION_ROTATE_180:
                    degree = 180;
                    break;
                case
ExifInterface.ORIENTATION_ROTATE_270:
                    degree = 270;
                    break;
            }
        }
    }
}

```



```

    }
    return degree;
}

private static String
convertStreamToString(InputStream is) {

    BufferedReader reader = new
    BufferedReader(new InputStreamReader(is));
    StringBuilder sb = new StringBuilder();

    String line = null;
    try {
        while ((line = reader.readLine()) !=
null) {
            sb.append((line + "\n"));
        }
    } catch (IOException e) {
        e.printStackTrace();
    } finally {
        try {
            is.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
    return sb.toString();
}
}

```

Kode Sumber 7.5. Kode MainActivity.java (Bagian 5)

```

public class CameraView extends SurfaceView
implements SurfaceHolder.Callback{

    private SurfaceHolder mHolder;
    private Camera mCamera;

    public CameraView(Context context, Camera
camera){
        super(context);
        mCamera = camera;
        mHolder = getHolder();
        mHolder.addCallback(this);
    }
}

```

```

mHolder.setType(SurfaceHolder.SURFACE_TYPE_NORMAL);
    }

    @Override
    public void surfaceCreated(SurfaceHolder
surfaceHolder) {
        try{
            mCamera.setPreviewDisplay(surfaceHolder);
            mCamera.startPreview();
            mCamera.setDisplayOrientation(90);
        } catch (IOException e) {
            Log.d("ERROR", "Camera error on
surfaceCreated " + e.getMessage());
        }
    }

    @Override
    public void surfaceChanged(SurfaceHolder
surfaceHolder, int i, int i2, int i3) {
        if(mHolder.getSurface() == null)
            return;

        try{
            mCamera.stopPreview();
        } catch (Exception e){
        }

        try{
            mCamera.setPreviewDisplay(mHolder);
            mCamera.startPreview();
        } catch (IOException e) {
            Log.d("ERROR", "Camera error on
surfaceChanged " + e.getMessage());
        }
    }

    @Override
    public void surfaceDestroyed(SurfaceHolder
surfaceHolder) {
    }
}

```

Code Sumber 7.6. Kode *CameraView.java*

```

<FrameLayout
xmlns:android="http://schemas.android.com/apk/res/an
droid"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".MainActivity">

    <FrameLayout
        android:id="@+id/camera_view"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:focusable="true"
        android:focusableInTouchMode="true" >

    </FrameLayout>

    <TextView
        style="@style/ButtonText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"

        android:textColor="@android:color/holo_green_light"
        android:textSize="24dp"
        android:textStyle="bold"
        android:layout_marginLeft="10dp"
        android:paddingTop="20px"
        android:id="@+id/objectTxt"
        android:layout_gravity="left|top" />

    <TextView
        style="@style/ButtonText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="40px"

        android:textColor="@android:color/holo_blue_bright"
        android:textSize="22dp"
        android:textStyle="bold"
        android:layout_marginLeft="10dp"
        android:paddingTop="30px"
        android:id="@+id/caloriesTxt"
        android:layout_gravity="left|top" />

```

```

<TextView
    style="@style/ButtonText"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="90px"

    android:textColor="@android:color/holo_blue_bright"
    android:textSize="22dp"
    android:textStyle="bold"
    android:layout_marginLeft="10dp"
    android:paddingTop="30px"
    android:id="@+id/fatTxt"
    android:layout_gravity="left|top" />

<TextView
    style="@style/ButtonText"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="120px"

    android:textColor="@android:color/holo_blue_bright"
    android:textSize="22dp"
    android:textStyle="bold"
    android:layout_marginLeft="10dp"
    android:paddingTop="50px"
    android:id="@+id/carbsTxt"
    android:layout_gravity="left|top" />

<TextView
    style="@style/ButtonText"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="160px"

    android:textColor="@android:color/holo_blue_bright"
    android:textSize="22dp"
    android:textStyle="bold"
    android:layout_marginLeft="10dp"
    android:paddingTop="60px"
    android:id="@+id/proteinTxt"
    android:layout_gravity="left|top" />

<ImageView
    style="@style/ButtonText"

```



```

        android:layout_width="75dp"
        android:layout_height="75dp"
        android:layout_marginBottom="50px"
        android:layout_marginLeft="60px"
        android:id="@+id/btn_identify"
        android:layout_gravity="left|bottom"
        android:src="@drawable/imgbtnpic"
        android:layout_above="@+id/textView"
        android:layout_alignParentStart="true"
        android:clickable="true" />

<ImageView
    style="@style/ButtonText"
    android:layout_width="75dp"
    android:layout_height="75dp"
    android:layout_marginBottom="50px"
    android:layout_marginRight="60px"
    android:id="@+id/btn_setting"
    android:layout_gravity="right|bottom"
    android:src="@drawable/setting"
    android:layout_above="@+id/textView"
    android:layout_alignParentStart="true"
    android:clickable="true" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="20px"
    android:layout_marginLeft="30px"
    android:textColor="@android:color/white"
    android:textStyle="bold"

    android:textAppearance="?android:attr/textAppearance
Small"
    android:text="IDENTIFY OBJECT"
    android:layout_gravity="left|bottom"
    android:id="@+id/identifyTxt"
    android:layout_alignParentBottom="true"
    android:layout_alignParentStart="true" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="20px"

```

```

        android:layout_marginRight="45px"
        android:textColor="@android:color/white"
        android:textStyle="bold"

        android:textAppearance="?android:attr/textAppearance
        Small"
        android:text="CONNECTION"
        android:layout_gravity="right|bottom"
        android:id="@+id/connectionTxt"
        android:layout_alignParentBottom="true"
        android:layout_alignParentStart="true" />

<ImageButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/imgClose"
    android:layout_gravity="right|top"

    android:background="@android:drawable/ic_menu_close_
    clear_cancel"
    android:padding="20dp" />

</FrameLayout>

```

Kode Sumber 7.7. Kode *Activity_main.xml*